3153/2

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
29 March 2001 (29.03.2001)

PCT

(10) International Publication Number
WO 01/22302 A1

(51) International Patent Classification[7]: G06F 17/50, 9/455, G06G 7/62

(21) International Application Number: PCT/US00/25808

(22) International Filing Date: 19 September 2000 (19.09.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

09/400,623	20 September 1999 (20.09.1999)	US
09/444,020	19 November 1999 (19.11.1999)	US
09/444,044	19 November 1999 (19.11.1999)	US

(71) Applicant: XIRCOM, INC. [US/US]; 2300 Corporate Center Drive, Thousand Oaks, CA 91320-1420 (US).

(72) Inventors: MONROE, Daniel; 1220 Mountain Meadows Drive, Woodland Park, CO 80963 (US). REA, Gregory; 3925 Barrelwood Court, Colorado Springs, CO 80920 (US). TURNER, Jeffery; 2130 Heatherdale Drive, Colorado Springs, CO 80915 (US). NEWMAN, Jeffrey; 8045 Sawback Trail, Colorado Springs, CO 80919 (US).

(74) Agents: WOLFF, Jason, W. et al.; Lyon & Lyon LLP, 633 West Fifth Street, Suite 4700, Los Angeles, CA 90071-2066 (US).

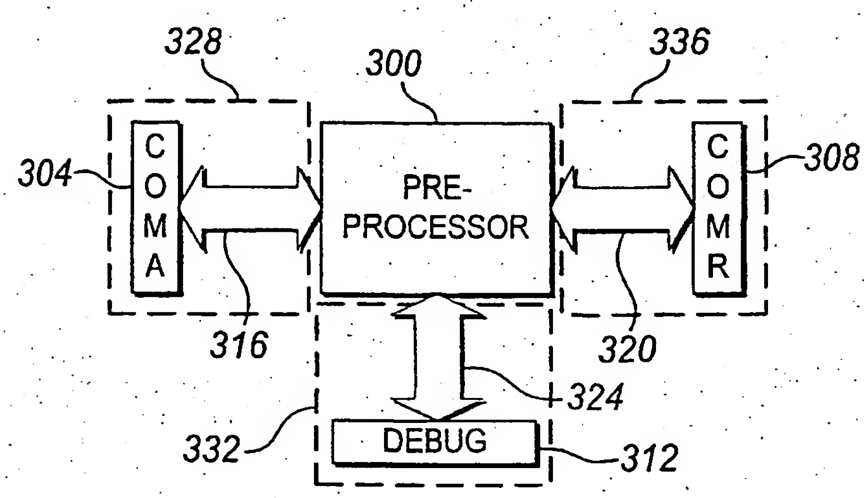
(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *With international search report.*
— *Before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: COMMUNICATIONS BRIDGE FOR CIRCUIT SWITCHED DATA TRANSFER SIMULATION



(57) Abstract: A communications bridge (300) for simulating circuit switched call links to user equipment receives data from the user equipment over a serial data interface (328). The communications bridge (300) simulates circuit switched call link responses back to the user equipment so that the user equipment believes that a circuit switched call link has been made. Data is sent through a second interface (336) to a wireless radio or GSM modem.

WO 01/22302 A1

Process this Application as a

PCT INTERNATIONAL APPLICATION

TITLE OF THE INVENTION

Communications Bridge for Circuit Switched Data Transfer Simulation

BACKGROUND

1. Field of Invention

This application relates to the field of communications equipment, and more particularly to communications equipment for integrating circuit switched and packet switched networks with a communications bridge.

2. Background Information

Monitoring and/or installed at residential, commercial, and industrial complexes of buildings throughout the world are a variety of user equipment. Examples of such user equipment include meter reading devices which measure consumption of various utility commodities such as natural gas, electricity and water via an electrical or electro-mechanical transducer. The meter reading devices are typically analog devices that record either a first reading and a second reading of the measured commodity over a period of time, or, alternatively, a cycling total (that is, a running total that recycles after a certain number is reached).

Generally, service personnel for the utility provider physically appear at or near the meter reading device to record consumption of the commodity each month. The recorded consumption from the meter reading device is then fed into a database

used for billing purposes which in turn generates an invoice for the consumer based on user's consumption of the measured commodity.

In urban areas, the number of meter reading devices that need to be recorded is tremendous. Although the overhead associated with sending service personal to a desired location can be amortized by consolidation of meter reading devices at a particular location, for example, in a high-density residential development such as an apartment complex, the cost can still be significant. In rural areas, however, the cost is higher as meter reading cannot generally be amortized over a number of meter reading devices read at a single location.

Various techniques are employed by utility companies to reduce the cost of sending service personal to a physical site.

For example, a simple method is the use of stochastic techniques for extrapolating a measured quantity for a current reading from one or more past values or a moving or seasonal average. This technique is designed to reduce the frequency of meter reading. A disadvantage, however, is the fact that the extrapolated reading can be greatly under or over the actual consumption, such as the case where a consumer is simply not present and no services are used, or when an unusual weather pattern occurs and consumption is significantly increased.

Another technique is the use of radio-based meter reading devices. For example, each meter reading device includes a radio, the radio capable of broadcasting a meter reading to a nearby receiver. In the Middle East for example, such a system is often employed because service personnel are frequently denied access to a property when the property owner (a man) is not home. The radio based meter reading devices allow service personnel to drive near the radio meter reading device with a receiver device to read the meter. With such technology service personnel do not need to enter the property. An advantage of such a system is that, in rural areas, the time it takes service personnel to read the meters can be reduced.

For example, one system might require service personnel to physically drive by or near a collection of meter reading devices in order to communicate with the devices. The data collected in the "drive-by" would be later uploaded to a centralized data collection system.

Another solution might include periodic stations that collect wireless data from the devices. The periodic stations, in turn could include a land-line modem that communicates with the centralized data collection system by way of circuit switched calls. Such a solution offers an alternative to deploying service personnel, however, setting up phone lines to service the periodic stations can also be expensive. Moreover, circuit switched calls can also be expensive.

For instance, one solution is to use GSM modems to make circuit switched calls. These GSM modems operate, for setup and control purposes, similarly to known landline modems, such as Hayes compatible modems. Such modems generally act primarily as a transport mechanism for moving data between two devices, such as a client computer and a head-end server. In particular, the modem is configured to passively modulate data or control signals from the client computer or to passively push modulated signals through the modem to the client computer.

When a modem is initialized, it will generally run either from a default modem configuration, which is stored in the modem, or the modem will receive a special initialization configuration parameters from the client computer. The initialization configuration is performed, in either mode, to ensure that the modem is able to communicate with the client computer and, preferably, operate (as a transport) at its maximum efficiency.

FIG. 16 shows a typical communication network 1600 employing a wireless modem. A laptop computer is the client computer 1604. The client computer 104 is connected, for example through an RS-232 port and serial cable 1606, to a GSM modem 1608. In turn, the GSM modem 108 is coupled to a GSM network 1612 via over-the-air interface 110. The GSM network 112 is coupled to a head-end server 1616, such as an application server, via a short message service center interface 1614.

As data flow 1650 illustrates, when a client computer 1604 tries to connect to the head-end server 1616, it will first try to initialize the GSM modem 1608 by passing configuration parameters directly to the GSM modem 1608. For example, configuration parameters 1620 are passed to the GSM modem 1608 over the serial cable 1606 from the client computer 1604. A response 1624, indicating whether the parameters were accepted, is then sent by the GSM modem 1608 back to the client

computer 1604. Once the GSM modem 1608 is initialized, it essentially becomes a slave to the client computer 1604 and functions as a passive transport carrying data 1628 from the client computer 1604 to the head-end server 1616, in addition to data 1632 from the head-end server 1616 to the client computer 1604.

The problems mentioned above are exacerbated by deregulation of the utilities industry in the United States. It is now possible for several different suppliers of electricity to service a single metropolitan area. This, in turn, results in a non-contiguous patchwork of service areas that service personnel may have to monitor. No longer can it be assumed that all users in a particular geographic area receive electrical power from a single service provider. Indeed, on a single residential block every household may have a different service provider. Moreover, the alleged ease with which a consumer may switch service providers further complicates the circumstance. Accordingly, the ability to amortize the costs of collecting usage measurements is reduced.

Other types of user equipment including vending machines, fire detectors, burglar detectors, and gas detectors. Typically, these devices need an alarm mechanism to indicate that a particular event has been detected. In most cases the alarm need only be audible or visible. However, often a response to a particular alarm condition requires intervention by a remote party. For example, when a fire alarm or burglar alarm is triggered it is useful to send the alarm to a central dispatching point so the appropriate response personnel can intervene.

The most commonly employed alarm system is used on many vending machines. In this alarm system, a service light one the vending machine is illuminated when an alarm is triggered. A sticker adjacent to the illuminated service light reads, "FOR SERVICE OR REPAIR PLEASE CALL JIM AT (408) 557-4567".

In more sophisticated environments, more complex alarm systems can be employed. For example, some devices utilize an internal microcontroller to detect an alarm condition and execute a response. The response is sent to a collection point and from there to an external landline modem, which utilizes its own microcontroller. The landline modem, in turn, notifies a dispatching center of the

alarm. An important aspect of this system is that the alarm system is under control of and/or triggered by the microcontroller internal to the device.

SUMMARY OF THE INVENTION:

A communications bridge for simulating circuit switched call links to user equipment is provided. According to one embodiment, the communications bridge receives data from the user equipment over a serial data interface. The communications bridge simulates circuit switched call link responses back to the user equipment so that the user equipment believes that a circuit switched call link has been made. Once the simulated circuit switched call link is established, data from the user equipment is packetized by the communications bridge into short message service or general packet radio service packets for transmission over a non-circuit switched call link.

An operator independent, transparent wireless modem management system is also disclosed. According to an embodiment, the modem management system comprises the steps of receiving a short message service message at the wireless modem and examining the short message service message for modem management information. If the short message service message comprises modem management information, then the wireless modem processes the short message service message.

In another embodiment, the memory of the communications bridge comprises a wireless protocol stack and a notification algorithm. An alarm condition received at the event detection interface triggers the notification algorithm. The notification algorithm processes the alarm condition, which preferably includes setting up an notification command that is passed to the wireless protocol stack. The wireless protocol stack in turn communicates a short message service message via the RF transceiver, the message including data that corresponds to the alarm condition.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 depicts a physical packaging of an embodiment of the inventions.

FIG. 2 depicts physical packaging of an alternative embodiment of the inventions.

FIG. 3 is a block diagram of a preprocessor unit and interface architecture.

FIG. 4 is a hardware schematic of the preprocessor unit.

FIG. 5 is a memory map of an embodiment of the inventions.

FIGS. 6A-C are block diagrams of embodiments of the inventions including a user equipment.

FIGS. 7A-B depicts state diagrams for two of the interrupt service routines of the inventions.

FIG. 8 is a flowchart depicting a main loop for a preprocessor driver.

FIGS. 9-12 are flowcharts depicting interrupt service routines for the preprocessor driver.

FIG. 13A is a block diagram of a system employing the inventions.

FIG. 13B is a flow diagram showing a simulated circuit switched call set-up.

FIG. 13C is a flow diagram showing a simulated circuit switched call tear down.

FIG. 14 is a block diagram of an network monitoring system employing the inventions.

FIGS. 15A-C depict various protocol stacks used in embodiments of the inventions.

FIG. 16 shows a typical communication network and data flow.

FIG. 17 shows a communication network and data flow according an embodiment of the invention.

FIG. 18 is a flowchart depicting the steps for a short message service modem management process.

FIG. 19A is a block diagram of an embodiment of an event detection and notification application for the wireless modem.

FIG. 19B depicts an external device that the event detection and notification modem can connect with.

FIG. 20 is a flowchart showing a main routine for the event detection and notification modem.

FIG. 21 is a flowchart showing a model notification algorithm for the event detection and notification modem.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

A method and apparatus for circuit switched data transfer simulation is provided. According to an aspect of the inventions, a communications bridge (or interface) is provided that deceives a data collection device into believing that circuit switched communications are being performed. In an embodiment, communications are actually performed by way of GSM short messaging services (hereinafter "SMS").

According to one embodiment of the inventions, the communications bridge is implemented by way of specially configured electrical hardware and software. However, according to another embodiment, the communications bridge is implemented by way of functionality added to an application layer of a GSM protocol stack on existing GSM modem hardware. Both embodiments are described herein.

OPERATIONAL OVERVIEW

FIG. 13A is a block diagram of one embodiment of a system 1300 employing the inventions. User equipment 1304 is a data collection device, such a meter reader collection station that receives data from a number utility meters. User equipment 1304 is communicatively coupled (e.g., by a serial data interface 1306) to a data terminal apparatus 1308, which functions as a communications bridge. Data terminal apparatus 1308 is configured to communicate with a GSM network 1320 over a wireless interface (or "over-the-air" interface) 1324, preferably by way of a standard GSM modem which is a component of the data terminal apparatus 1308. For now, the left side of the GSM network 1320 will be called, for convenience, the originator equipment. The commands described below are preferably implemented from a modified Hayes AT command set.

To the right of GSM network 1320, and also linked by a wireless interface 1328, is a similar data terminal apparatus 1312 which is also communicatively coupled (e.g., by a serial data interface 1314) to user equipment 1316. User equipment 1316 can include a second data collection station or other device for analyzing or relaying communications from the originator equipment. For convenience, the right side of the GSM network 1320 will be referred to as receiver

equipment. Note that receiver equipment does not have to mirror the originator equipment, for example, the receiver equipment does not have to be coupled to the GSM network 1320 by way of a wireless interface 1328 and can instead be coupled by way of physical network connections.

FIGS. 13B-C depict a call setup and call tear down protocol for circuit switched call simulation. According to one embodiment, neither the originator user equipment 1304 nor the receiver user equipment 1316 will be aware that a non-circuit switched call was made. The data terminal apparatuses 1308 and 1312 simulate circuit switched call response to the user equipment and thus make the fact that a non-circuit switched exchange was performed transparent.

Turning first to FIG. 13B, it is aligned with FIG. 13A and depicts a flow diagram for a simulated circuit switched call setup. Starting from user equipment 1304, an ATD command 1332 is issued and serially passed to the data terminal apparatus 1308 over the serial data interface 1306. The communications bridge handles the incoming ATD command 1332 and sends an SMS establish link message 1336 to the wireless radio. According to one embodiment, the data terminal apparatus and user equipment negotiate flow control so as to prevent input buffer overflows from data being transferred from the user equipment 1304 to the data terminal apparatus 1308.

The wireless radio transmits the SMS establish link message 1336 over the wireless interface 1324 to the GSM network 1320. The GSM network 1320 routes the SMS establish link message 1336 to the receiver wireless local loop 1328. At the receiver wireless local loop 1328, the SMS establish link message 1336 is then routed to data terminal apparatus 1312, which receives the message at its wireless radio and then handles the message with its communications bridge. The communications bridge examines the message and notifies the user equipment 1316 of an incoming call with a ring indicator 1340.

The data terminal apparatus 1308 communications bridge preferably keeps the phone number active for five minutes. This is to accord sufficient time to receive an acknowledgment of the SMS establish link message 1336 from the user equipment 1316.

According to one embodiment, the AT command "ATA" (shown as ATA command 1344) is passed from the user equipment 1316 to the data terminal apparatus 1312. The data terminal apparatus 1312 then sends an SMS link established message 1348 to the data terminal apparatus 1308. Upon receipt of the SMS link establish message 1348, the data terminal apparatus 1308 communications bridge sends a connect message 1352 to the user equipment 1304.

Note that if more than one SMS establish link message 1336 is received from the GSM network 1320 by the data terminal apparatus 1312 before an ATA command 1344 is received from the user equipment 1316, then the data terminal apparatus 1312 communications bridge responds to the most recent SMS establish link message 1336.

Once the SMS link is established, data can be transferred between the user equipment 1304 and the user equipment 1316 over via short messaging services routed over the wireless local loop(s) by the data terminal apparatuses 1308 and 1312.

With a virtual link between the originator equipment and the receiver equipment, data can be passed as if a regular circuit switched call is being performed. The operation is transparent to the user equipment, as the data terminal apparatus communications bridge handles all data packetization, handshaking, sequencing and error correction required by the particular application in which the equipment is employed.

Now turning to FIG. 13C, it is also aligned with FIG. 13A. FIG. 13C depicts a call link tear down 1350 flow diagram for a simulated circuit switched call. According to one embodiment, the communications bridge in the data terminal apparatus 1308 waits approximately ten minutes for data or commands from the user equipment 1316. If no data or commands are received in such time frame, then the call is considered "dropped". However, receiving an escape sequence also causes the call to be dropped.

First, the escape sequence 1356 is received by the data terminal apparatus 1308. An SMS disconnect link message 1360 is then transmitted over the wireless interface 1324 by the wireless radio in data terminal apparatus 1308.

The GSM network 1320 receives the transmitted SMS disconnect link message 1360 and routes it over wireless interface 1328 to data terminal apparatus 1312. Data terminal apparatus 1312 receives the SMS disconnect link message 1360 and it is processed by the communications bridge. The communications bridge, in turn, sends a disconnect indicator 1364 to the user equipment 1316 and then a link disconnected message 1368 back to data terminal apparatus 1308. When the link disconnect message 1368 is received by the data terminal apparatus 1308, the wireless radio drops the link.

Although described above with reference to an SMS embodiment, according to another embodiment, the communications bridge simulates circuit switched calls by way of general packet radio services ("GPRS"). The call setup and tear down are substantially similar to the methods described above (and below), however, rather than supplying a phone number after AT command "ATD", an internet protocol address is supplied (e.g., "ATD114.32.0.108"). Once a connection is established, data packets passed over GPRS are formatted similar to the data packets over SMS.

Another difference between the SMS and GPRS approaches is that rather than having a 140 byte packet length and a baud rate of less than 300 baud (SMS), GPRS packets can have a 1500 Kbytes packet length, moreover, a much higher over-the-air rate, 170 Kbaud, is also possible. The addition of larger data packets and increased bandwidth allows easier integration of additional functionality into the communications bridge, such as relaxed flow control, handshaking for improved quality of service, multi-bit encryption, and other error recovery techniques (e.g., parity checks, CRC, etc.). While such features are possible in the SMS embodiment, the small packet size may require packet sequencing number and other header information to be sent with each packet, which would further slow communications.

One example of a system implementing the general architecture described above is a home automation application running on a personal computer (user equipment 1304), that interfaces data terminal apparatus 1308 via interface 1306. At the opposite end of the home automation application resides a home network control center embodied in user equipment 1316. The home automation application provides monitoring and control services to the home network control system, whereas the home network control system controls, for example, heating,

ventilation, air conditioning, and security for a user's home. Commercially available home network control systems include Echelon Corporation's LONWORKS™ technology.

Another example of a system implementing the general architecture depicted in FIG. 13A is an automatic meter reading system. In such a system, user equipment 1304 is an automatic meter reader collection station that receives measurements of consumption of a metered commodity, such as electrical power or natural gas, for one or more automatic meter readers. Data measured by the automatic meter readers is sent to the collection station where it is in turn fed to the data terminal apparatus 1308. User equipment 1316 can be a utility device that either actively polls the collection station via data terminal apparatus 1312, or passively receives measured data from the data collection station via data terminal apparatus 1312.

PHYSICAL PACKAGING

FIG. 1 depicts a perspective view of an embodiment of the physical packaging of a data terminal module ("DTM") 100. The data terminal module 100 includes an enclosure 104 that surrounds a data terminal sub-assembly ("DTSA"). The data terminal sub-assembly (not shown) is a circuit card that is configured to receive a preprocessor and wireless radio, which are described in detail below.

A coaxial cable connector 108, a standard DB-9 connector 112, a power connector 116 and a power indicator 120 are shown on the on the enclosure 104. Each is connected, internally, to the data terminal sub-assembly. The coaxial cable connector 108 is configured to receive an antenna for the wireless radio. Two mounting sleeves 106 are notched into the enclosure 104. The mounting sleeves 106 provide a path for connectors that are used to secure the data terminal module 100 to a desired location.

FIG. 2 depicts a perspective view of an embodiment of the physical packaging of a data terminal unit ("DTU") 200. The data terminal unit 200 includes a two-part enclosure. Case 204 carries electronics modules, such as data terminal sub-assembly 228 and power supply 232. A heat sink 224 is placed at each inside corner of case 204 and is used to dissipate heat generated by the electronics modules. The outside surface of case 204 comprises a coaxial cable receptacle 240,

and an A/C power cord 244. Also shown on the outside surface of case 204 are two hinged latches 216.

The second part of the enclosure for data terminal unit 200 is a cover 208. Cover 208 is connected to case 204 via hinges 248, and is configured to sealably enclose the electronics modules carried in the case 204. Latch connectors 212 receive hinged latches 216 to assist in this end. The cover 208 and the case 204 also include a number of connector receptacles 220 for additional protection. A patch antenna 236 is mounted to the cover 208. The patch antenna 236 is coupled to the wireless radio contained in the data terminal sub-assembly 228.

Details of a patch antenna 236 and embodiments of an enclosure are described in U.S. patent application Serial Nos. 09/316,457, entitled "CAPACITIVE SIGNAL COUPLING DEVICE", and 09/316,459, entitled "RADIATING ENCLOSURE", both filed May 21, 1999.

According to an alternative embodiment, special electrical hardware is not employed in either the data terminal module 100 or data terminal unit 200. In such an embodiment, application software is added to a standard GSM modem software stack. Accordingly, the data terminal apparatuses can be a specially configured GSM modem. However, in another embodiment specially configured electronics and/or memory can be employed. One such embodiment is now presented.

PREPROCESSOR EMBODIMENT

FIG. 3 is a block diagram of the preprocessor architecture. Preprocessor 300 is coupled to a first interface 328, a second interface 336, and a third interface 332. Preferably, each of the interfaces includes an RS-232 port having a DB-9 or equivalent physical connector. COMA 304, COMR 308 and DEBUG 312, for example, can be implemented with such connectors. Communication lines 316, 320, and 324 communicatively couple the physical connectors to the preprocessor 300.

According to one embodiment, the first interface 328 connects to user equipment (e.g., telemetry equipment, automatic meter reading equipment, meter reader concentration point, utility meter control system, substation monitoring equipment, etc.). The user equipment is configured to collect measured data that monitors external activity. The second interface 336 is a physical connection to a

wireless radio, more specifically a GSM modem having a baud rate of approximately 9600 bps or higher. The third interface 332 is preferably an open serial interface capable of receiving a terminal or test equipment for debugging and configuration purposes. According to one embodiment, the debug port services can be physically accessed through the first interface 328.

Each RS-232 connector 304, 308 and 312 is shown coupled to the preprocessor 300 by unique communication lines 316, 320 and 324. This is for simplicity and to represent a unique address for each communication port or serial interface. In fact, a single address and data bus can support the communication ports.

FIG. 4 is a hardware schematic an embodiment of the preprocessor 300. The preprocessor 300 comprises a microcontroller 404, preferably Dallas Semiconductor part no. DS80C323 (16 MHz), a universal asynchronous receiver transmitter ("UART") 416, preferably an Exar Corporation part no. ST16C2450 (8 MHz), a non-volatile memory 424, preferably Advanced Micro Devices part no. 29LV001B-70JC, and a volatile memory 428, preferably IDT part no. 71V256SA-12PZ. Additional control logic 420 is desired, such as gate arrays and TTL logic, for maintaining timing (e.g., a clock divider for the UART 416), buffering, and logic levels. Power circuitry 412 provides power to the preprocessor 300 and any peripheral device (e.g., a wireless radio), and a crystal oscillator 408 (16 MHz) provides a clock signal. A main bus 432 communicatively couples the microcontroller 404, with memories 424 and 428, as well as control logic 420. The main bus 432 includes both data, address and control lines, such as the same control lines 436 interconnecting the clock 408, the UART 416 and the microcontroller 404.

Additional lines 440, 444 and 448 are shown connected to UART 416. These lines are for the first interface 328, second interface 336, and third interface 332. Interrupts are received by the UART 416, over lines 440, 444, and 448, which trigger exception/interrupt algorithms in the microcontroller 404. A portion of the volatile memory 428 is used as a 1024 byte memory buffer for each input queue in UART 416 (thus, if two interfaces are used, 2048 bytes of memory are used).

FIG. 5 shows a memory map for 65 kilobytes of address space. The lower 49 kB address space 504 is for the non-volatile memory 424, the next 12 kB of

address space 508 is for volatile memory 428, followed by 8 bytes of address space 512 for the first interface 328, followed by 2 kB of address space 516 of reserved memory, 8 bytes of address space 520 for the second interface 336, and another 2 kB of address space 524 for reserved memory.

FIGS. 6A-C depict various physical embodiments of the inventions interfaced with user equipment (e.g., data collection unit 608). FIG. 6A shows a single user equipment 600 comprising a data collection unit 608 (e.g., an automatic meter reader), a preprocessor 604 and a wireless radio 612. The preprocessor 604 is coupled to the data collection unit 608 via a first interface 616, and to the wireless radio 612 via a second interface 620.

FIG. 6B shows a user equipment 624 including the preprocessor 604. Here, wireless radio 612 is in a separate physical packaging 628. Here, the second interface comprises I/O interfaces 652 and 656 (e.g., RS-232 ports). The I/O interfaces 652 and 656 are connected via a serial cable 644. I/O interface 656 is coupled to preprocessor 604 via connector 640, and I/O interface 652 is coupled to wireless radio 612 via connector 648.

FIG. 6C shows a user equipment 632, which is similar to the system depicted in FIGS. 6A and 6B, however, the preprocessor 604 and wireless radio 612 are found in data terminal unit/module 636. The same interface described above with reference to FIG. 6B and the second interface 620 is shown in FIG. 6C. However, in FIG. 6C the components found in the second interface 620 are found in the first interface 616 instead. FIG. 6C is most like the data terminal module 100 and data terminal unit 200 shown in FIGS. 1 and 2 respectively.

An operational overview of the techniques are now presented. Generally speaking, the preprocessor 604 is a hardware component that includes a software driver. As described above, the preprocessor 604 can have a dedicated piece of hardware that executes the software driver, however, it is also possible for the software driver to be overlaid into an existing piece of hardware as an additional component of the software stack. For example, the preprocessor driver can be added to the user equipment software stack or to the wireless radio software stack. The preprocessor driver, as it is referred to herein, is generally an interrupt driven service routine that first identifies the source of an interrupt and second determines what

process or interrupt service routine to execute based on any of the data accompanying the interrupt (or the interrupt itself).

It should be noted that the incoming data is preferably serial ASCII character data. Commands are preferably based on the well-known Hayes modem AT Command set, although additional special codes can be added to identify particular functionality described herein. Some of these codes are described below with references to Table 2.

According to one embodiment, the primary components of the preprocessor driver are saved in non-volatile memory 424 (e.g., section 504 of memory map 500) and are executed by microcontroller 404 as a sequence of instructions stored in a computer-readable format. For example, the sequences of instructions (e.g., op codes) are loaded into data and control registers within the microcontroller 404 from the non-volatile memory 424 (alternatively, the instructions can be copied from non-volatile memory 424 to a volatile execution memory before being executed). The sequences of instructions cause the microcontroller 404 in the preprocessor 300 to perform a series of acts based upon a combination of the sequences of instructions and the data received from the serial data interface (e.g., UART 416). Program variables needed by the preprocessor driver are either stored in available registers internal to the microcontroller 404, or they are stored in volatile memory 428.

Two state diagrams are described below with reference to FIGS. 7A-B. The descriptions are general and are further supported by the detailed flowcharts described below with reference to FIGS. 8-12.

A state diagram 700 for the first interface interrupt service routine is shown in FIG. 7A. The default state is IDLE mode 704. In the IDLE mode 704, characters are received over the first interface 616 and tested for commands or events that trigger a state change, for example link commands 716 and 728. If a command or event indicating a state change is not detected then the data characters are stored in a memory buffer until a state change does occur.

The circuit ("CKT") mode 712 passes data from the first interface 616 directly to the second interface 620, with little intervention from the preprocessor 604, except to monitor for commands or events that may trigger another state change (e.g., an escape sequence). In the circuit mode 712, an active virtual link is

maintained between the wireless radio 612 and a public telephony switched network ("PSTN") over a wireless local loop. The circuit mode 712 is maintained until an escape sequence 720 is detected. However, if a re-enter command 732 is detected following the escape sequence 720, then the interrupt service routine will return to the circuit mode 712 and not to IDLE mode 704.

The short message service ("SMS") mode prepares and sends SMS messages comprising the data stored in the memory buffer from the first interface 616, over the second interface 620, and through the wireless radio 612. From the wireless radio 612, the SMS messages are carried over the GSM network and are then routed over other intervening networks to their ultimate destination.

SMS messages generally have a 140 byte data structure. The first byte indicates an SMS message type, the second byte indicates the SMS message length, and the last 138 bytes comprise the SMS message body. The SMS message body comprises either character data, or commands, or both. The SMS messages types are described in Table 1.

Table 1

establish link	requests that a virtual link be established for SMS data transfer
link established	a reply to an establish link message indicating that the link is setup (the sending or receipt of this message causes the mode to change to SMS mode)
data link	all data is transferred using this message type
disconnect link	requests that a link be disconnected (the sending or receipt of this message causes the mode to change to IDLE mode)
link disconnected	a reply to the disconnect link message

An escape sequence 724 causes the state to return from SMS mode 708 to IDLE mode 704.

State diagram 750, shown in FIG. 7B, depicts the states associated with the second interface interrupt service routine. The states described above with reference to the first interface interrupt service algorithm are substantially similar to those associated with the second interface interrupt service algorithm, the primary difference being that if an interrupt was received over the second interface, then it is

not from the user equipment (e.g., data collection 608), but rather from remote equipment beyond the wireless radio 612.

In IDLE mode 754, data characters received at the second interface 620 (e.g., COMR 308) are passed straight through the first interface 616 (e.g., COMA 304). The data characters are, however, monitored for a link command 758 or 762, which indicate a state change to SMS mode 766 or CKT mode 770, respectively. Escape sequences 774 and 778 return the service routine to IDLE mode 754 from CKT mode 770 or SMS mode 766.

FIG. 8 depicts the main loop 800 for the preprocessor driver. The main loop 800 begins by first initializing various operating parameters. For example, in act 804, a watchdog timer, a second timer, the serial ports, and the UART 416 are initialized.

The watchdog timer is designed to reset the preprocessor 604 in the event that the software stalls or a processing error occurs. Preferably, the duration of the watchdog timer is set to 4.5 seconds. A second timer is used to extend the 4.5 second timeout for routines that take longer than the first watchdog timer. The second timer generates a 2 millisecond interrupt.

The serial ports (e.g., COMA 304 and COMR 308) are initialized to operated at 9600 baud, 8 data bits, 1 stop bit, and no parity, and the UART 416 is initially setup to run at 9600 baud. Memory buffer input queues have 1024 bytes each and store data characters received through COMR 308 and COMA 304.

Next, in act 808, the mode of the main loop is set to IDLE (e.g., for both the first interface 616 and the second interface 620). After act 808, the interrupt service routine processing begins.

In act 812, if an interrupt was received at the first interface 616, then an exception occurs and processing continues to the first interface interrupt service routine in act 816, which is described above with reference to FIG. 7A and below with reference to FIG. 9.

In act 820, if an interrupt was received at the second interface 620, then an exception occurs and processing continues to the second interface interrupt service routine in act 824, which is described above with reference to FIG. 7B and below with reference to FIG. 10.

In act 828, a test is performed to determine whether the elapsed time since the last data character was received over the first interface 616 (e.g., through COMA 304) when the first interface interrupt service routine is in SMS mode 708. The test is referred to as the SMS timer expire event. If the SMS timer event has occurred, then in act 832 the memory buffer is prepared for SMS transmission, the SMS message is transmitted and the SMS timer is reset. Processing continues to act 836.

In act 836, if an interrupt was received at the third interface (e.g., debug interface 312 depicted in FIG. 3) then the debug interrupt service routine described below with reference to FIG. 11 is performed in act 840.

In act 844, a test is performed to determine whether values of a modem status register or a line status register (taken from wireless radio 612 by UART 416), have been updated. If either has been updated, then processing continues to act 848, described below with reference to FIG. 12. After, alternatively, acts 844 or 848, the process continues to act 852, at which point the loop is restarted at act 812.

FIG. 9 is a flowchart depicting the acts performed by the preprocessor driver when servicing an interrupt over the first interface 616. More specifically, the flowchart depicts the first interface interrupt service routine 900, which services interrupts from the user equipment (e.g., data collection 608).

In act 904, a test is performed to determine whether the present mode is CKT mode 712. If the mode is CKT mode 712, then the data received by the first interface 616 is passed through to the second interface 620 at act 908.

In act 912, a test is performed to determine whether an escape sequence was received. In particular, the test determines whether the character sequence "+++" was received through the first interface 616. If the escape sequence was not received, then in act 916 the escape sequence counter is reset. However, if the escape sequence was received, then in act 920, the mode is set to IDLE mode 704. The preprocessor driver then returns to the main loop, namely act 820.

In act 924, a test is performed to determine whether the present mode is IDLE mode 704. If the present mode is IDLE mode 704, then the incoming data character from the first interface 616 is added to a memory buffer. Next, in act 932, a test is performed to determine whether a process trigger (e.g., a carriage return, a CTRL-Z, or the memory buffer is full) has occurred. If a process trigger has

occurred, then in act 936, the memory buffer is parsed, interpreted and the appropriate acts performed. A list of exemplary interpreted strings and their results is shown in Table 2.

Table 2

AT-EMU	sets non-volatile memory to reflect the simulation mode -- 0 indicates no simulation, 1 indicates simulated circuit switch
ATDnnnnnnn	sends an establish link SMS message to phone number nnnnnnn (when AT~EMU set to 1)
ATH	sends a disconnect link SMS message to the phone number established using the ATD command (when AT~EMU set to 1)
ATS7?	returns the value 30 to the first interface 616
AT0	returns to CKT mode if the carrier detect signal on the wireless radio 612 is still asserted
AT+IPR=xxxx	changes the baud rate on the wireless radio 612 and on both ports on the preprocessor 604 to xxxx (2400, 9600)
AT?	displays this table to the debug port

It is noted that the AT~EMU command sets the simulation mode for the unit. This is important because the unit is capable of providing both true circuit switched calls or simulated circuit switched calls. Accordingly, if the mode is set to no emulation, then the ATD and ATH commands will operate as setup/control functions for a truly circuit switched call. However, when the mode is set to simulated circuit switched mode, the functions are unique -- as described above.

After act 936, the preprocessor driver returns to act 820.

In act 940, a test is performed to determine whether the present mode is SMS mode 708. If the present mode is SMS mode 708, then in act 944 the SMS timer is reset. In act 948, another test is performed to determine whether there is sufficient room in the memory buffer to store additional data. If there is not sufficient room, then in act 952 an SMS message is sent thereby flushing a portion of the memory buffer. If, however, there is sufficient room in the memory buffer, then in act 956 any escape characters are handled, for example the occurrence of the string "+++" through COMA 304, or a disconnect link SMS message received through COMR 308. After handling the escape characters, the preprocessor driver returns to act 820.

FIG. 10 is a flowchart depicting the acts performed by the preprocessor driver when servicing an interrupt over the second interface 620 – i.e., the second interface interrupt service routine 1000.

In act 1004, a test is performed to determine whether the present mode is CKT mode 770. If the present mode is CKT mode 770, then any data characters received at the second interface 620 (e.g., COMR 308) are passed through the preprocessor 604 to the first interface 616 (e.g., to COMA 304) in act 1008. Next, in act 1012 a test is performed to determine whether an escape sequence has been received. According to one embodiment, the escape sequence is the receipt of three consecutive plus signs, or the string "+++". If the escape sequence is not found, then the escape counter is reset in act 1016. However, if the escape sequence is found, then the mode is set to IDLE mode 754 in act 1020. After acts 1016 or 1020, the preprocessor driver continues to act 828.

In act 1024, a test is performed to determine whether the present mode is IDLE mode 754. If the present mode is IDLE mode 754, then data characters received at the second interface 620 are passed through to first interface 616 in act 1028. In act 1032, a test is performed to determine whether a link sequence is found in the data characters. According to one embodiment, the link sequence is the string "+CMTI:", which indicates an incoming SMS message. If the link sequence is not detected, then the escape counter is reset at act 1036. However, if a link sequence was detected, then the inbound SMS message is read in act 1040.

Next, in act 1044, a second test is performed to determine whether another link sequence is found in the SMS message (e.g., in the SMS message type field). For example, the next link sequence can be the "establish link" or "link established" messages described above with reference to Table 1. If no establish link or link established messages is found, then the remainder of the SMS message is output to the first interface 616 in act 1048. However, if the establish link or link establish message is found, then the message is processed and the mode is set to SMS mode 766 in act 1052. The preprocessor driver then continues to act 828.

In act 1056, a test is performed to determine whether the present mode is SMS mode 766. If the present mode is SMS mode 766, then a test is performed in act 1060 to determine whether a parse trigger has been received. According to one

embodiment, parse triggers include a line feed or a carriage return, as well as a "buffer full" indicator. If no parse trigger has been received, then in act 1064 the data character is stored in the memory buffer and processing continues to act 828. However, if a parse trigger has been received, then in act 1068, the SMS message is parsed.

In act 1072, a test is performed to determine whether the SMS message type is disconnect link. If the SMS message type is not disconnect link, then in act 1076, the SMS message type is sent to the first interface 616. However, if the SMS message is a disconnect link command, then the mode is set to IDLE mode 754 in act 1080. After step 1080, the preprocessor driver continues to act 828.

FIG. 11 is a flowchart depicting the debug interrupt service routine 1100. Under normal circumstances a third interface, shown in FIGS. 3 and 4, receives the debug interrupt. The debug interrupt is commonly associated with connecting a terminal device or laptop computer into the third interface 332. The debug interrupt service routine 1100 is used for setup and diagnostic purposes.

In act 1104, a data character received over the third interface 332 is added to the memory buffer. In act 1108, the data character is tested to determine whether it is a carriage return. If the data character is not a carriage return, then the routine returns to act 844. If the data character is a carriage return, then in act 1112 the memory buffer is parsed. In act 1116, a test is performed to determine whether a command from the memory buffer is in a command list (e.g., Table 2). If the command is not in the command list, then in act 1120 an error message is reported over the third interface 332. However, if the command is in the command list, then the command is executed in act 1124. Thereafter, processing continues to act 844.

FIG. 12 depicts a flowchart 1200 for handling the modem status register ("MSR") and line status register ("LSR") values gathered from the first interface 616 and second interface 620. Essentially, a series of possible errors are analyzed and the appropriate action is taken, which includes, in some instances changing the preprocessor mode.

In act 1204, a test is performed to determine whether the carrier detect ("CD") is asserted on the second interface 620 (e.g., the wireless radio 612). If the

carrier detect is asserted, then the carrier detect is asserted on the first interface 616 and the mode is set to the circuit mode in act 1208.

In act 1212, a test is performed to determine whether the carrier detect is dropped on the second interface 620. If the carrier detect is dropped, then in act 1216 the carrier detect is then dropped on the first interface 616 and the mode is set to the idle mode.

In act 1220, a test is performed to determine whether the ring indicator ("RI") is asserted on the second interface 620. If the ring indicator is asserted, then in act 1224 the ring indicator is asserted on the first interface 616.

In act 1228, a test is performed to determine whether the ring indicator is dropped on the second interface 620. If the ring indicator is dropped, then in act 1232 the ring indicator is dropped on the first interface 616 as well.

In act 1236, a test is performed to determine whether the clear-to-send ("CTS") is asserted on the second interface 620. If the clear-to-send is asserted, then in act 1240 the wireless radio 612 is initialized. After acts 1236 or 1240, the preprocessor returns to act 852.

The service routine embodied in flowchart 1200 is useful in that the service routine can change the mode of the preprocessor 604 at either the first interface 616 or the second interface 620, in response to certain physical conditions, namely the assertion or dropping of one of the lines used to complete a virtual circuit.

APPLICATION LAYER EMBODIMENT

According to another embodiment, special purpose simulation hardware (e.g., preprocessor 300 shown in FIG. 3) is not integrated with an existing GSM modem. Rather, the functionality described above is implemented by way of software added to the application layer of a standard GSM protocol stack 1500 shown in FIG. 15A. The same software can also be added to the general packet radio service (GPRS) protocol stacks 1580 (GPRS Class C) and 1590 (GPRS Class A), shown in FIGS. 15B and 15C, respectively. Like reference numerals in FIGS. 15A-C refer to like elements.

According to an embodiment, the call setup and tear down functionality described above with reference to FIGS. 13A-C is performed by way of extensions

to the application layer of the GSM protocol stack, for example, the GSM protocol stack already existing in a GSM modem.

In such an embodiment, computer-readable program code is compiled and loaded into a non-volatile storage medium. The code is later executed by one or more processors configured to handle the incoming AT commands from the user equipment or the wireless input in the form of SMS packets. The application layer extensions build a functional communications bridge for simulating circuit switched calls to the user equipment. An advantage of the application layer embodiment is that no special/single purpose hardware is required. Rather, functionality is added to an existing GSM modem by way of the software added to the GSM protocol stack.

FIG. 15A depicts a GSM protocol stack 1500. The base components of GSM protocol stack 1500 are generally known in the art. For example, the base software components of GSM protocol stack 1500 are available from various venders such as debis Systemhaus in Berlin, Germany, CONDAT Datensystem Gmblt in Hannover, Germany, and other wireless communications vendors. According to one embodiment, the GSM protocol stack 1500 can be implemented in a variety of logic devices or in computer readable code executed by an embedded processor already part of the GSM modem.

The inventions can be embodied in software code that comprises the AT command interface 1504. The AT command interface is overlaid onto each of the various GSM protocol stacks 1500, 1580 and 1590. Commands, as described above with reference to Table 2, are thus bridged between the user equipment 1304 and the GSM protocol stack -- and, hence, the GSM network 1320 (FIG. 13). It is further noted that the AT command interface 1504 can also include event detection and notification software that detects alarms from the user equipment and handles them appropriately -- for example by initializing a simulated circuit switched or circuit switched call. Some exemplary AT commands are shown in Table 3.

Table 3

+++AT~LOGON=[password]	logs a user on to the wireless modem and allows other mobile management functions to be executed
+++AT~LOGOFF	logs a user off the wireless modem or disables mobile management functions
+++AT~PWD=[password]	changes the existing password
+++AT[command]	modem management command identifier

Referring to FIG. 15A, the user equipment 1304 sends data over a GSM network 1320 using the Hayes standard AT command interface 1504. The mobile network man machine interface (MN) 1512 receives the data and passes the data to the appropriate messaging service -- e.g., a short message service (SMS) 1516, a call control service (CC) 1520, or a supplementary service (SS) 1524. A registration element 1508 will provide the mobility management layer 1528 with necessary information about the data and the network. From each of layers 1508, 1516, 1520 and 1524 data flow is then directed to and from the mobility management layer (MM) 1528.

The mobility management layer 1528 establishes, maintains, and releases connections between the user equipment 1304 and the GSM network 1320. From the mobility management layer 1528, data and control is passed to the radio resource management layer (RR) 1532. The radio resource management layer 1532 establishes physical connections over the radio interface for call-related signaling and traffic channels between the user equipment 1304 and base station 1488 (FIG. 14).

Connected to the radio resource management layer 1532 is the physical layer (L1) 1540. The physical layer 1540 processes call-related signaling and traffic channels directly from the radio resource layer 1532, and also processes the data sent from the data link layer (L2) 1536.

FIG. 15B is substantially similar to FIG. 15A, however, the "G" notation in the protocol stack layers indicates that the designated layers now refer to a general packet radio service (GPRS). GPRS uses a packet radio principle and can be used

for carrying packet data protocol between the user equipment 1304 and the GSM network 1320. GPRS provides additional services beyond what is offered with the standard GSM network, for example, GPRS can provide increased over-the-air data transfer rates and packet lengths.

An application program interface (API) 1544 is added to allow an application to control the subnetwork dependent convergence protocol (SNDCP) 1548, which is responsible for segmentation and re-assembly of the data packets, encryption and decryption, and transmission control protocol (TCP) header and data compression.

Layers interfacing the AT command interface 1504 include the registration layer 1508 and man-machine interface layer 1512, which in turn interface the SM layer 1552 and GSMS (GPRS short message service) 1556. The SM layer 1552 and GSMS layer 1556 interface the GPRS mobility management (GMM) layer 1560, and both the GMM and SNDCP layer 1548 interface the link layer control (LLC) 1564, which handles the link layer information of the packet data.

Link layer control 1564 interfaces the GPRS resource management layer (GRR) 1568. GPRS resource management layer 1568 in turn interfaces medium access control / radio link control (RLC/MAC) layer 1572, which handles the physical link processing, as well as physical layer 1540.

FIG. 15C shows the GPRS Class A protocol stack 1590. The protocol stack 1590 is a merge of the GSM protocol stack 1500 (FIG. 15A) and the GPRS Class C protocol stack 1580 (FIG. 15B), which is denoted by the dual reference numbers annotating the various layers of the protocol stack. The GPRS Class A protocol stack can operate standard GPRS and other GSM services simultaneously.

DATA COLLECTION SYSTEM

FIG. 14 depicts an alternative system 1400 employing the invention. User equipment 1404 comprises an application program, for example a telemetry, automatic meter reading, meter concentration point, utility meter control system, substation monitoring, home network control system, or other application. In particular, the inventions can be used in conjunction with an event detection and notification application, such a fire alarm, gas alarm, burglar alarm, vending

machine alarm, or another condition indicating a change of state of the user equipment 1404, or some other device connected thereto.

An RS-232 interface 1448 with hardware flow control connects user equipment 1404 to data terminal module 100, or alternatively data terminal unit 200. Data terminal module 100 and data terminal unit 200 preferably comprise a debug port through which terminal equipment or a laptop computer 1412 can interface and perform installation or testing services with software tools 1484. Optionally, the debug and configuration service can be accessed through the first interface 616.

The data terminal module 100 and data terminal unit 200 communicate via a wireless radio to one or more antenna relays 1408. Preferably the wireless radio is a GMS type modem. The wireless radio is configured to transmit and receive information between said data terminal module 100, or data terminal unit 200 and GSM network 1488.

At least one of the one or more antennas relays 1408 is connected to a base transceiver station ("BTS") 1416. The base transceiver station 1416 processes the inbound wireless data (e.g., forms data packets for the inbound wireless data) and routes it over a T1 line 1456 (or other leased line) to base station controller ("BSC") 1420. The base station controller 1420 authenticates service for the data terminal unit/module and directs the processed inbound wireless data over T1 line 1456 to a mobile switching center ("MSC") 1424. The mobile switching center 1424 directs the processed inbound wireless data over a lease line 1456 to an appropriate networking station, for example an interworking function ("IWF") 1432, such as a PSTN bridge/router in the case of a circuit switched data path (identified by label "1"), or a short message service center ("SMSC") 1428, in the case of a short message service data path (identified by label "2").

If the data path is a circuit switched data path, then from the interworking function 1432 the processed wireless data is passed over a public switched telephony network ("PSTN") 1436 to circuit switched data interface 1436. If, however, the data path is a short message service data path, then the short message service center 1428 can route the processed wireless data over a PSTN connection 1460 to interface 1436, or over an packet switched network 1456 connected to an internet 1440. In the case of routing over the internet 1440, the short message service center

1428 handles all Internet Protocol packetization according to known Internet Protocol standards, such as publicly available Internet RFC 791, which is incorporated herein by reference in its entirety.

A user application server 1444 retrieves the inbound wireless data from the PSTN interface 1436 via a modem connection 1468. Alternatively, the user application server 1444 retrieves the inbound wireless data via an internet access/service provider 1476 connected to internet 1440.

Optional terminal management software 1480 can be used by the user application server 1444 to provide outgoing data, command, or setup services from the user application server 1444 to the user equipment 1404 (though the data terminal module 100 or data terminal unit 200) in a reverse path as is described above.

OPERATOR INDEPENDENT, TRANSPARENT WIRELESS MODEM MANAGEMENT TECHNIQUES

The invention further comprises methods and techniques for operator independent, transparent wireless modem management using short message services. The methods and techniques are embodied in a wireless modem, although aspects of the techniques can be implemented in a remote (or "head-end") server. The techniques preferably allow for the wireless modem to function not only as a slave device to a client computer, as in the past, but as an intelligent processing device.

Furthermore, the wireless modem is configured to allow remote initialization of configuration parameters via short message service messages. The short message service messages can contain data that not only initializes the wireless modem's operating parameters, but data that calls other commands or applications executing internally or externally to the wireless modem. For example, the methods and techniques allow remote initialization of event detection and notification algorithms internal to the wireless modem. In another example, the methods and techniques allow remote initialization or control of a device external to the wireless modem, for example a utility meter, a vending machine, or other user equipment.

FIG. 17 depicts an exemplary network 1700 that employs the techniques. User equipment 1720 is connected via a serial cable or other line 1706 (hereinafter

"line 1706") to a wireless modem 1704, preferably a global system for mobile communications (hereinafter "GSM") or general packet radio service (hereinafter "GPRS") -type modem. Although user equipment 1720 is depicted as a laptop computer, it can be a less sophisticated or more limited functionality form of user equipment such as a utility meter, a meter concentration point, a utility meter control system, a substation monitor, telemetry equipment, or a vending machine.

The wireless modem 1704 is coupled to a wireless network 1612, for example a GSM or GPRS network, via an over-the-air interface 1610. The wireless network 1612 is in turn coupled through a short message service center interface 1614 to a head-end server 1616. The head-end server 1716 differs from head-end server 1616 in that it is configured to send the SMS messages to the wireless modem 1708, the SMS messages including the wireless modem management information. No special programming is required, however, on the head-end server 1716, as only the ASCII strings that comprise a standard SMS message need to be modified (as will be apparent in the description below, in particular with reference to Table 1).

A data flow diagram 1750, also in FIG. 17, shows possible communications flows between the wireless modem 1704 and the user equipment 1720, or the head-end server 1716. It is noted that data and configuration parameters can now pass to or from both the user equipment 1720 and the head-end server 1716. For example, data and configuration parameters 1724 can be passed from the client computer 1720 to the wireless modem 1704. Similarly, data and configuration parameters 1728 can pass from the wireless modem 1704 to the client computer 1720. Even more significantly, data and initialization parameters 1712 can be sent from the head-end server 1716 to the wireless modem 1704, just as data and initialization parameters 1708 can be sent from the wireless modem 1704 to the head-end server 1716. By overlaying the short message service management methods and techniques described herein, the wireless modem 1704, in effect, can operate as an intelligent bridge between the user equipment 1720 and the head-end server 1716, thereby facilitating communication and control between legacy devices and a wireless network.

FIG. 18 depicts an embodiment of a short message service modem management algorithm 1800. In step 1804, an SMS message is received at the

wireless modem 1704 originating from the head-end server 1716. The inbound SMS message is parsed in step 1808 and the results of the parse are stored in a memory of the wireless modem 1704. In step 1812, the data parsed from step 1808 is tested for a known command or indicator to determine whether the SMS message should be processed internally by the wireless modem 1704, or externally by the user equipment. For example, by comparing the stored data to a command lookup table or a particular sequence of characters (e.g., "+++AT") that indicate the SMS message is for modem management purposes.

If, as a result of testing in step 1812, a modem management command is recognized, then the SMS message is processed by a particular routine in the wireless modem in step 1816. However, if testing in step 1812 indicates that the SMS message should be passed through the wireless modem 1704, then in step 1820 the SMS message is passed through to the user equipment 1720.

After either step 1816 or step 1820, the initial processing terminates. It is noted, however, that processing likely involves further steps or specific functions that do not involve the actual handoff or identification of inbound SMS messages. Since such specific processing steps are generally beyond the scope of the invention, they are not further detailed hereinafter, rather, in step 1816 it is simply noted that the wireless modem processes the SMS message appropriately. In most cases, the same routines that handle data or commands from the user equipment 1720 are used to handle data and commands passed from the head-end server 1716 by way of SMS messaging.

For example, additional commands can include directing a processor in the wireless modem 1704 to check the quality of an RF signal detected at an RF transceiver, or to select a particular RF channel over the over-the-air interface will communicate certain information with the wireless modem. Further still, the commands can include directing a processor to retrieve a call history file or log that describes the calls sent from or received by the wireless modem 1704.

EVENT DETECTION & NOTIFICATION APPLICATION

FIG. 19A is a block diagram of electronic hardware in a wireless modem, substantially similar to the wireless modem depicted and described with reference to

FIG. 4 that is capable of handling the event detection and notification application of the invention. Wireless modem 1900 includes power and clock circuitry 1904. Preferably a power and ground plane are part of a printed circuit board that supports the electronic hardware. The clock signal is shown connected to a processor 1908, although it can also be provided to the other elements comprising modem 1900.

A non-volatile memory 1912, for example an electrically alterable but persistent memory such as flash memory or EEPROM, is communicatively coupled to the processor 1908. The non-volatile memory 1912 preferably holds software that is executed by the processor 1908 including, but not limited to GSM communications protocol stack software and notification algorithm software, which is described in further detail below. A volatile memory 1916, such as a random access memory, is also coupled to the processor 1908. The volatile memory 1916 preferably holds temporary program variables and other information used in a run-time environment, that is, when processor 1908 is executing the GSM protocol stack software or the notification algorithm.

A RF radio, or RF transceiver 1920 is also coupled to the processor 1908. The RF transceiver 1920 handles physical modification of data and control signals for transmission to or reception from the over-the-air interface 1922. Over-the-air interface 1922 communicates with a base station which is part of a GSM or GPRS network. General wireless communication standards such as GSM and GPRS are known in the art and thus it is not necessary that they be described in further detail here.

According to one embodiment, the processor 1908 includes one or more interrupt inputs. When a state transition is detected on an interrupt input, such as interrupt input 1924, a memory address (commonly referred to as an exception vector) of a routine that handles the interrupt is retrieved. The processor 108 then temporarily halts processing of a main routine so that the algorithm identified by the exception vector can be executed. As shown in FIG. 19a, an external signal is supplied over line 1932 to a receptacle 1928, which is in turn connected to the interrupt input 1924. The receptacle 1928 to interrupt input 1924 elements can be collectively referred to as an "event detection interface". It is worth noting that

other embodiments of event detection interfaces can also be employed, some of which are described in further detail below.

For the purpose of explanation, the external signal on line 1932 will be hereinafter referred to as an "alarm condition". The alarm condition can be virtually any type of alarm generated by external circuitry, for example, the alarm condition can be an active high/low signal indicative of a gas leak, a maintenance indicator (such as a vending machine running out of stock), or a circuit breaker tripping. It is noted that the line 1932 can comprise a single wire or multiple wires.

FIG. 19B shows a vending machine 1940. An input panel 1944 is used to select stock (such as a candy bar or a beverage) behind a display window 1948, stock that is, once paid for and selected, deposited at an opening 1952. When a particular stock or all stock stored in the vending machine 1940 reaches a minimum level, or when the vending machine runs out coins for making change, circuitry within the vending machine 1940 creates an alarm condition on line 1932. The circuitry can generate a high or low voltage strobe, a wave, a steady-state value, or a control sequence on the line 1932. The alarm condition on line 1932 is fed into the receptacle 1928 on the modem 1900, thereby triggering an exception vector that causes a notification algorithm to be executed by the processor 1908.

In an alternative embodiment, the alarm condition on line 1932 does not need to be sent to an interrupt input on processor 1908. Rather, the alarm condition can be fed on line 1932 to another input port associated with the processor 1908, such as an RS-232 port, a parallel port, or a particular data line on a parallel port. (All of the aforementioned devices are embodiments of event detection interfaces.) Furthermore, rather than triggering an exception vector that calls the notification algorithm, the processor 1908 can periodically poll the event detection interface for a change of state in the line 1932. If a state change is detected, then the main routine itself calls the notification algorithm.

FIG. 20 depicts an embodiment of the main routine 2000. The primary function of the main routine 2000 is to handle the wireless protocol and process data to and from the RF transceiver 1920. Accordingly, the main routine 2000 is essentially a wireless protocol stack, such as GSM protocol stack, a general packet

radio service ("GPRS") protocol stack, a mix of GSM and GPRS, or anther type of protocol stack, such as code division multiple access ("CDMA").

Step 2004 shows the main routine handling/executing the GSM processing. This processing will continue uninterrupted until an alarm condition (or "event") is detected in step 2008. If an alarm condition is detected, then processing continues to the notification algorithm 2100 depicted in FIG. 21.

Before explaining FIG. 21, it is important to note that more than one alarm condition, and thus more than one event notification algorithm can be programmed into the modem 1900. For this reason, FIG. 21 generally describes the notification algorithm's process, which can be applied to virtually any specific notification algorithm.

Now turning to FIG. 21, the alarm condition is identified in step 2104. In the case where the alarm signal is fed directly to a particular interrupt input, the fact that the interrupt input received the alarm signal is sufficient to determine which notification algorithm is to process the alarm condition. This is so since each interrupt input has its own exception vector (although the exception vectors may be the same as between two or more interrupt inputs). In step 2108, the notification algorithm 2100 sets up the particular notification command that will be transmitted using the wireless protocol. Step 2108 can further include a step of reading command data from a setup file that includes specific data or parameters used by the notification command.

In one embodiment, the notification will be transmitted in the form of a GSM short message service (hereinafter "SMS") message. Accordingly, a text script for transmission is created, such as "gas leak detected, IAC Hampton site, unit no 3, 07-23-99 05:34 PST". Similarly, the text script can be "out of stock B3, SCU building 5, unit no 2, 06-09-99 18:23 PST". In still another embodiment, multiple alarm conditions can be collected and summed by the notification algorithm 2100 each time the interrupt is detected. Once a threshold number of alarm conditions have been reached, for example six snack selections in a vending machine are empty, then the notification algorithm 2100 passes data along to the main routine 2000 for processing and delivery via the wireless protocol stack.

In one embodiment, the alarm condition triggers a standard command format. For example, the standard command format can be "AT~EVNTMSG=<num>,<"msg">", where num is the telephone number (or IP address as the case may be) to which a GSM SMS (or GPRS SMS) message is to be delivered and msg is the ASCII text script of the message. After the command is setup in step 2108, in step 2112 it is passed on to the main routine 2000 for processing. In the main routine 2000, the wireless protocol stack, which monitors for a notification command, receives the internally invoked command and executes it appropriately, for example by sending the msg text to the appropriate telephone number (num) as a SMS message.

The text scripts and other data used as parameters for the event message command described above are preferably stored in non-volatile memory (such as memory 1912) so that the commands only have to be programmed once and that no processing requirements (beyond the alarm condition circuitry – which likely already exists) are placed on the external device (for example vending machine 1940).

Where a number of external events are to be detected by the modem 1900, a standard demultiplexor circuit in the external device can generate the alarm condition that is fed over line 1932 to the modem 1900. In such an embodiment, the processor 1908 needs to monitor the entire port and differentiate between interrupts/inputs (as step 2104 indicates).

As for the wireless protocol stack that is implemented in the main routine 2000, FIGS. 15A-C depict various embodiments of wireless protocol stacks that include SMS messaging. A preferred GSM protocol stack is shown in FIG. 15A, while a general packet radio service (hereinafter "GPRS") protocol stack is shown in FIG. 15A. FIG. 15B depicts a combined GSM and GPRS protocol stack. Like reference numerals in FIGS. 15A-C refer to like elements.

According to one embodiment, the AT command interface 1504 and SMS layer 1516 are modified in the protocol stack so that the main routine 2000 will recognized notification commands from the notification algorithm 2100 (for example, the command "AT~EVNTMSG=<num>,<"msg">" that was described above with reference to FIG. 21). In particular, the AT command interface 1504

provides a convenient layer in which such special purpose commands can be integrated without requiring substantial modification of the wireless protocol stack, while still leaving the flexibility to integrate additional functionality.

An embodiment of the invention is preferably embodied in software code that comprises the AT command interface. The AT command interface is overlaid onto each of the various wireless protocol stacks. The AT command interface preferably receives a notification command from the notification algorithm 2100, the notification command being triggered by an alarm condition set by circuitry in an external device.

A typical data flow is as follows. Circuitry in an external device, such as vending machine 1940, detects an event. The circuitry triggers an alarm condition on line 1932. Line 1932, fed into the modem 1900 detects the alarm condition at its event detection interface. The alarm condition in turn causes the processor 1908 to enter an exception/interrupt processing mode to halt processing in a main routine 2000 and to call a special purpose notification algorithm 2100. The notification algorithm 2100 in turn processes the alarm condition and then creates a notification command and data that is passed to the AT command interface. From the AT command interface the command and data will be handled by the remainder of the wireless protocol stack.

The main routine 2000 (which includes the wireless protocol stack) and the notification algorithm 2100 can be implemented by way of one or more sequences of instructions stored in a computer-readable medium, such as a non-volatile memory communicatively coupled to one or more processors in the wireless modem. The sequences of instructions can be precompiled (as in the case of object code), or they can be interpreted instructions. Furthermore, the main routine 2000 and notification algorithm can be implemented in a variety of programming languages, such as C, C++, assembly, or BASIC. As for the notification commands that are passed between the notification algorithm 2100 and main routine 2000, they can be implemented by the same programming languages, but preferably they are based on a modified AT command set.

Furthermore, whereas the main routine 2000 and notification algorithm 2100 are generally stored as executable object code, the setup and initialization strings

(e.g., the phone number, IP address, text message, and number of alarm conditions that trigger an event message) for particular event notification commands passed to the AT command interface can be stored as one or more modifiable text files. A debug port, an additional port, or even the event detection interface (e.g., an RS-232 port) on the wireless modem can allow for programmically modifying the setup and initialization strings, or they may be programmically modified by sending an SMS or other type message from a head-end server to the wireless modem.

An advantage of the invention is that the methods and apparatuses described herein can be incorporated into a single data terminal apparatus or wireless modem and then integrated with legacy equipment that is in need of event detection and notification technology. The invention, when so incorporated, thereby reduces the overhead associated with employing service personnel to check on such equipment. Further, it provides an off-the-shelf, single piece of hardware that can be added on to a legacy external device while requiring minimal, if any, modifications to the external device. Further still, the event detection and notification processing (if any) in the legacy external device can be moved out of the external device and onto a convenient, multi-purpose wireless modem.

CLAIMS

What is claimed is:

1. A communications bridge, including a first interface configured to receive data over a serial connection and translate said data into packet data for transmission by a wireless radio over a GSM network, said communications bridge further configured to simulate a circuit switched call link to a user equipment through said serial data connection.

2. The communications bridge of claim 1, wherein said packet data is in short message service format.

3. The communications bridge of claim 1, wherein said packet data is in general packet radio service format.

4. The communications bridge of claim 1, further comprising an application layer object code added to a GSM protocol stack, said application layer object code configured to handle said simulated circuit switched call link to said user equipment.

5. The communications bridge of claim 1, further comprising:

a microcontroller;

a non-volatile memory coupled to said microcontroller;

a volatile memory coupled to said microcontroller; and

an input output controller coupled to said microcontroller, said input output controller including:

said serial data connection, and

a second serial data connection configured to communicatively couple to said wireless radio.

6. A method for simulating a circuit switched call link comprising:

establishing an simulated circuit switched call link with a user equipment;

preparing data received over said simulated circuit switched call link for transmission as one or more packets over a non-circuit switched call link;

monitoring said data received over said simulated circuit switched call link for an escape sequence; and

tearing down said non-circuit switched call link and said simulated circuit switched call link if said escape sequence is detected.

7. The method of claim 6, further comprising preparing packet data received via a non-circuit switched call link into serial data for transmission over said simulated circuit switched call link.

8. The method of claim 6, wherein preparing said data received over said simulated circuit switched call link includes formatted said data into short message service format packets.

9. The method of claim 6, wherein preparing said data received over said simulated circuit switched call link includes formatted said data into general packet radio service format packets.

10. The method of claim 6, further comprising:

sending flow control data to said user equipment, said flow control data indicating a capacity of said data terminal apparatus to handle data flows from said user equipment; and

configuring status information for verifying successful transmission of data to a destination.

11. A computer readable medium having stored therein one or more sequence of instructions for simulating a circuit switched call link, said one or more sequences of instructions causing one or more processors to perform the acts of any of the above claims 6-10.

12. A wireless modem comprising:

a processor;

a memory coupled to said processor, said memory comprising a software stack configured to hold one or more sequences of instructions for causing said processor to handle a wireless protocol;

a RF transceiver coupled to said processor; and

an interface coupled to said processor, said interface configured to receive signals from an external device; wherein

said processor is configured to handle said signals from said external device over said interface, and wherein said processor is further configured to handle one or more short message service messages delivered over said RF transceiver, said short message service messages comprising modem management information.

13. The wireless modem of claim 12, wherein said modem management information comprises wireless modem configuration parameters.

14. The wireless modem of claim 12, wherein said modem management information includes a command for wireless modem to perform a function stored internally to said wireless modem.

15. The wireless modem of claim 12, wherein said processor handles said one or more short message service messages received over said RF transceiver by parsing data from one or more said short message service messages, testing said parsed data for a modem management command indictor, and processing said short message service message if said modem management command indicator is detected.

16. The wireless modem of claim 12, wherein said wireless protocol is a global system for mobile communications protocol.

17. The wireless modem of claim 12, wherein said wireless protocol is a general packet radio services protocol.

18. A method for wireless modem management comprising the steps of:

receiving a short message service message at a wireless modem;

examining said short message service message for modem management information;

processing said short message service message at said wireless modem if said short message service message comprises said modem management information; and

passing said short message service message through said wireless modem if said short message service message does not comprise said modem management information.

19. The method of claim 18, said step of examining comprising the steps of:

parsing said short message service message;

testing said parsed short message service message for a modem management command indicator, said command indicator indicating whether said short message service message includes said modem management information; and

signaling to a processor in said wireless modem that a command is waiting to be executed if said processing designator is found.

20. The method of claim 18, wherein said step of processing said short message service message at said wireless modem includes initializing said wireless modem based upon said modem management information contained in said short message service message.

21. The method of claim 18, wherein said step of processing said short message service message at said wireless modem includes the step of checking a quality of a wireless signal detected at said wireless modem.

22. The method of claim 18, wherein said step of processing said short message service message at said wireless modem includes the step of handling a request for a call log history.

23. The method of claim 18, wherein said step of processing said short message service message at said wireless modem includes the step of selecting a RF channel for wireless communications on said wireless modem.

24. The method of claim 18, wherein said step of processing said short message service message at said wireless modem includes the step of authenticating a party sending short message service messages to said wireless modem.

25. The method of claim 18, wherein said step of processing said short message service message at said wireless modem includes the step of initializing communication parameters for event detection and notification.

26. A computer-readable medium comprising one or more sequences of instructions configured to cause a processor to perform steps for modem management in a wireless modem, as recited in any of above claims 18-25.

27. A modem comprising:

a processor;

a memory coupled to said processor, said memory comprising a software stack configured to hold one or more sequences of instructions for causing said processor to handle a wireless protocol;

a RF transceiver coupled to said processor; and

an event detection interface coupled to said processor, said event detection interface configured to receive a signal indicative of an alarm condition; wherein

said processor is configured to handle said signal indicative of said alarm condition by executing a notification algorithm from said software stack, said notification algorithm sending data to said RF transceiver.

28. The modem of claim 27, wherein said wireless protocol includes global standard for mobile communications short message service messaging.

29. The modem of claim 27, wherein said wireless protocol includes general packet radio services short message service messaging.

30. The modem of claim 27, wherein said signal indicative of said alarm condition is directed to an interrupt input on said processor.

31. The modem of claim 27, wherein said signal indicative of said alarm condition is detected by said processor periodically polling said event detection interface.

32. The modem of claim 27, wherein said event detection interface is communicatively coupled with an external device, said external device including circuitry for generating said signal indicative of said alarm condition.

33. A method for event detection and notification using a wireless modem comprising the steps of:

executing a wireless protocol in a processor enclosed in said wireless modem;

receiving a signal indicative of an alarm condition at an event detection interface coupled to said processor; and

executing a notification algorithm in said processor in response to said step of receiving said signal indicative of said alarm condition.

34. The method of claim 33, wherein said wireless protocol includes global standard for mobile communications short message service messaging.

35. The method of claim 33, wherein said wireless protocol includes general packet radio services short message service messaging.

36. The method of claim 33, further comprising the steps of:

polling said event detection interface by said processor for said alarm condition at periodic intervals; and

calling said notification algorithm if a particular alarm condition is detected.

37. The method of claim 33, further comprising the step of generating an exception vector in response to said step of receiving said signal indicative of said alarm condition, wherein said exception vector causes said notification algorithm to be called.

38. A computer-readable medium comprising one or more sequences of instructions configured to cause a processor to perform steps for event detection and notification in a wireless modem, said one or more sequences of instructions comprising the steps of any of the above claims 33-37.

39. A wireless modem comprising:

a processing means;

a memory coupled to said processing means, said memory comprising a wireless communication protocol stack; and

a RF transceiver coupled to said processing means; wherein the improvement comprises:

an event detection interface coupled to said processing means, said event detection interface configured to receive an alarm condition from an external device; and

wherein if said alarm condition is received at said event detection interface, then executing a notification algorithm, also stored in said memory, said notification algorithm invoking a notification command that triggers a short message service message to be passed to said RF transceiver.

40. The wireless modem of claim 39, wherein said wireless protocol includes global standard for mobile communications short message service messaging.

41. The wireless modem of claim 39, wherein said wireless protocol includes general packet radio services short message service messaging.

42. The wireless modem of claim 39, wherein said signal indicative of said alarm condition is directed to an interrupt input on said processor.

43. The wireless modem of claim 39, wherein said signal indicative of said alarm condition is detected by said processor periodically polling said event detection interface.

44. The wireless modem of claim 39, wherein said event detection interface is communicatively coupled with an external device, said external device including signal generation means, said signal generation means configured to generate said signal indicative of said alarm condition.

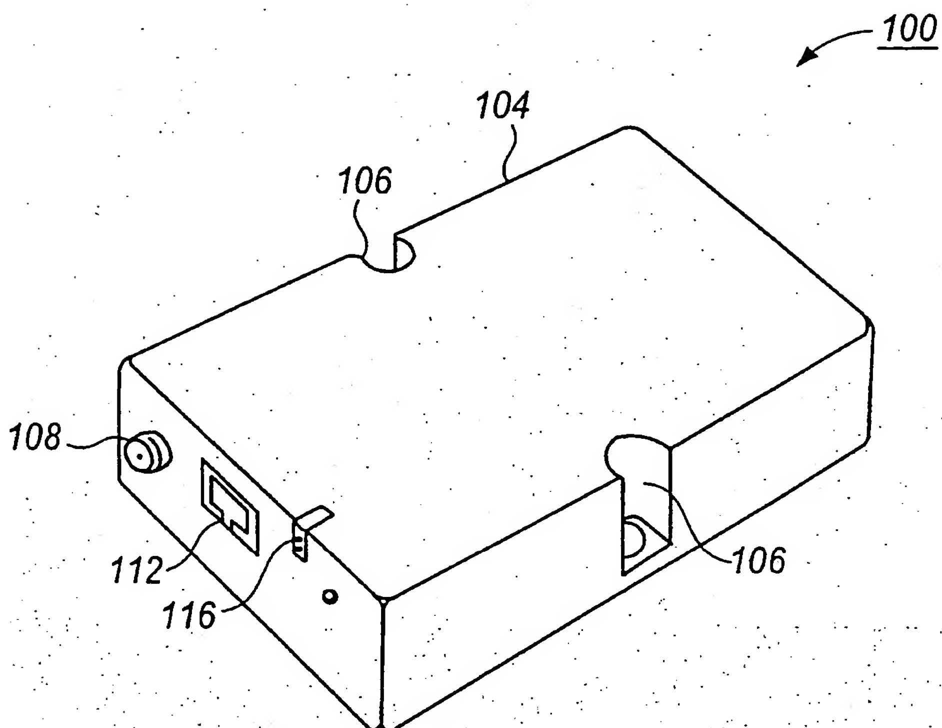


FIG. 1

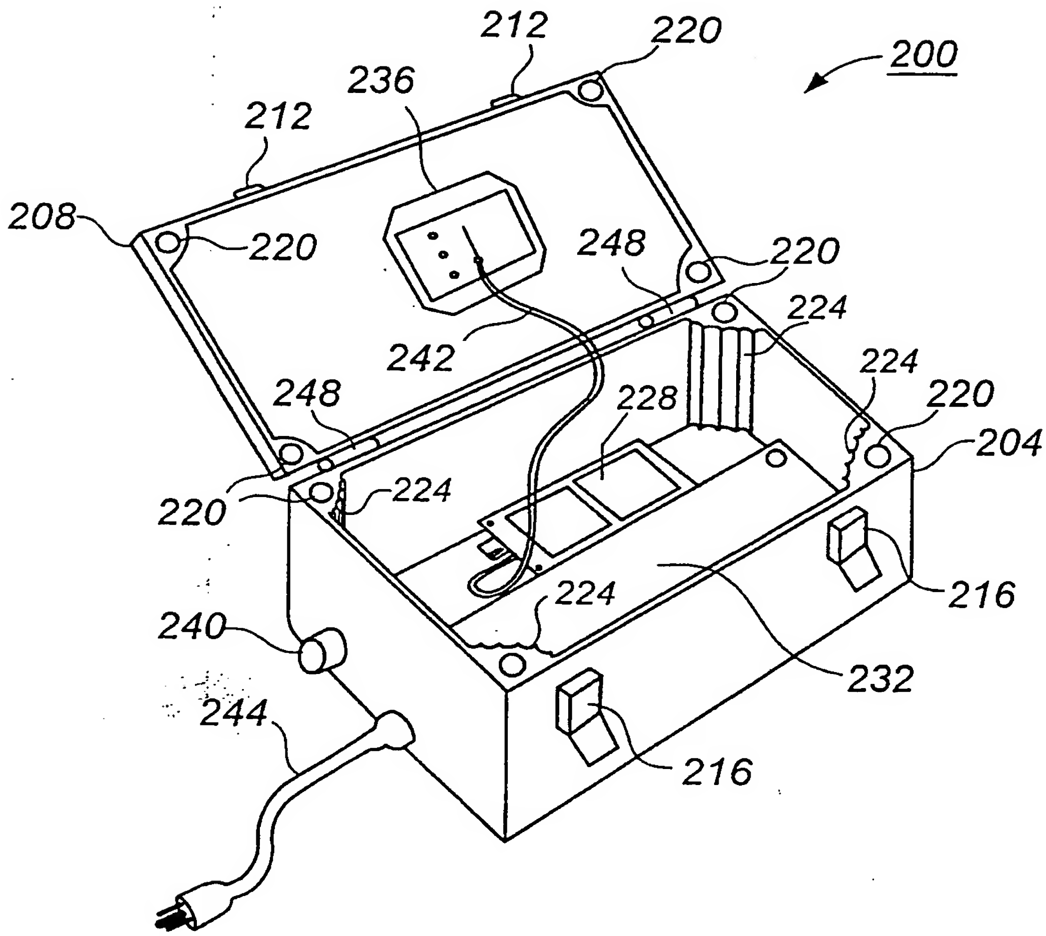

212
220
200
236
212
208
220
248
220
224
242
224
248
228
220
204
220
224
216
224
240
232
244
216

FIG. 2

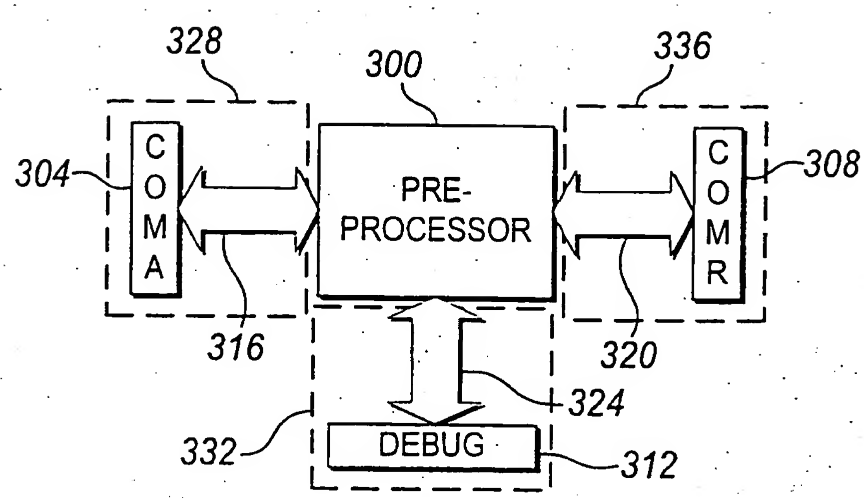

328
300
336
304
COMA
PRE-
PROCESSOR
COMR
308
316
320
324
332
DEBUG
312

FIG. 3

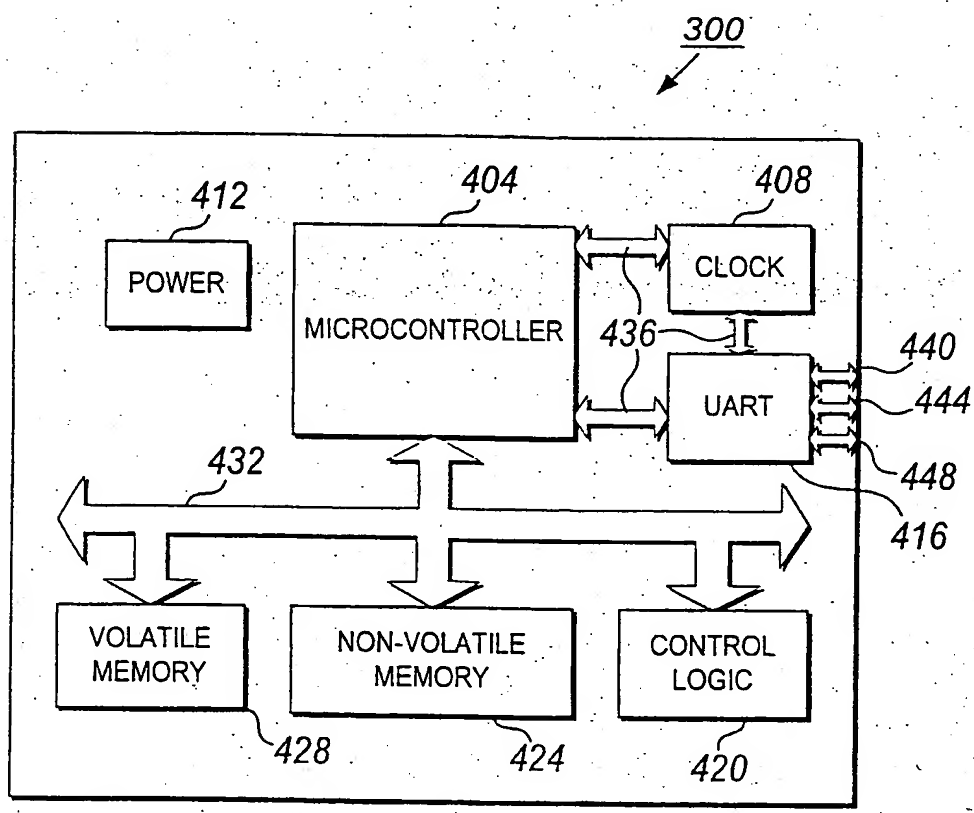

300
412
404
408
POWER
MICROCONTROLLER
CLOCK
436
UART
440
444
448
432
416
VOLATILE
MEMORY
NON-VOLATILE
MEMORY
CONTROL
LOGIC
428
424
420

FIG. 4

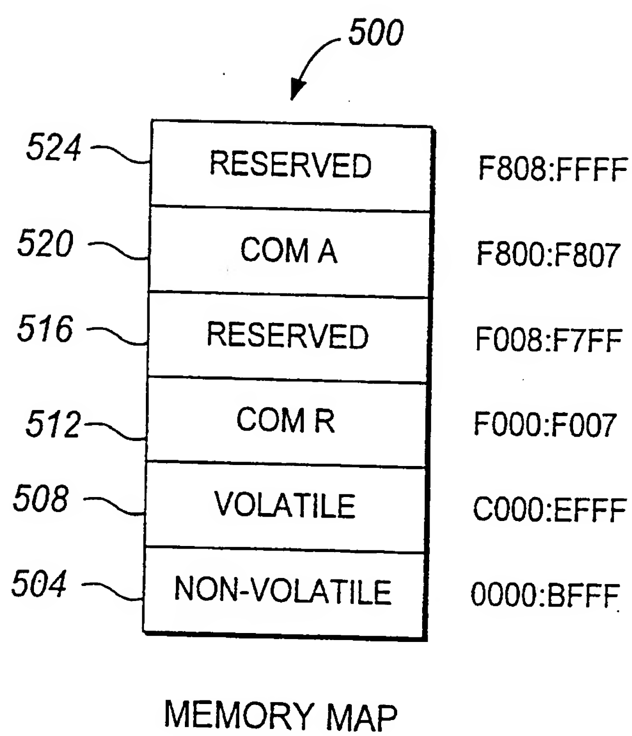

500
524
RESERVED
F808:FFFF
520
COM A
F800:F807
516
RESERVED
F008:F7FF
512
COM R
F000:F007
508
VOLATILE
C000:EFFF
504
NON-VOLATILE
0000:BFFF
MEMORY MAP

FIG. 5

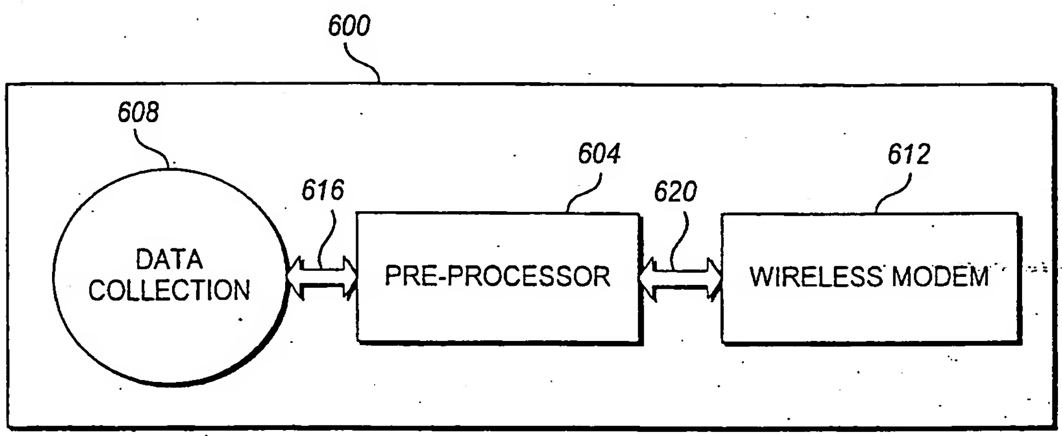


FIG. 6A

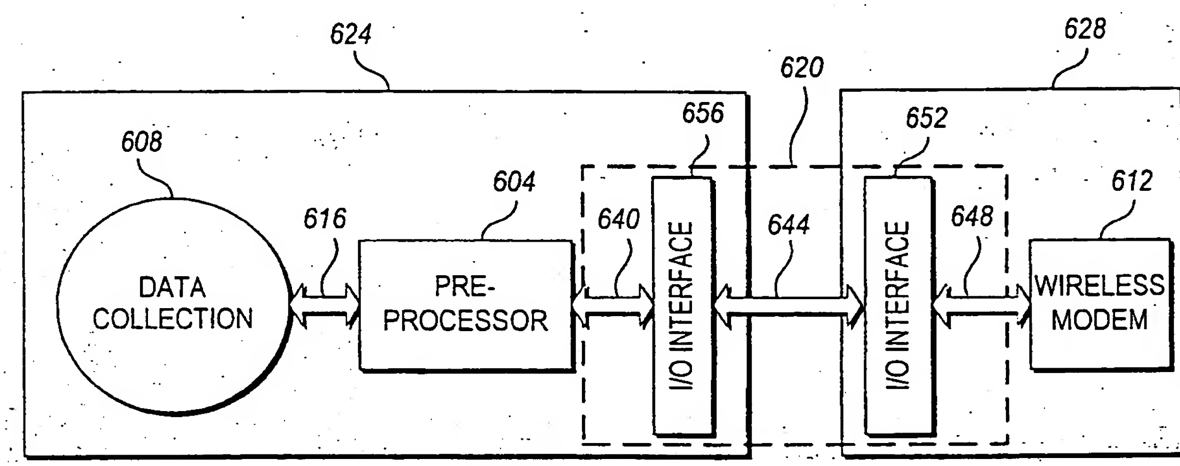


FIG. 6B

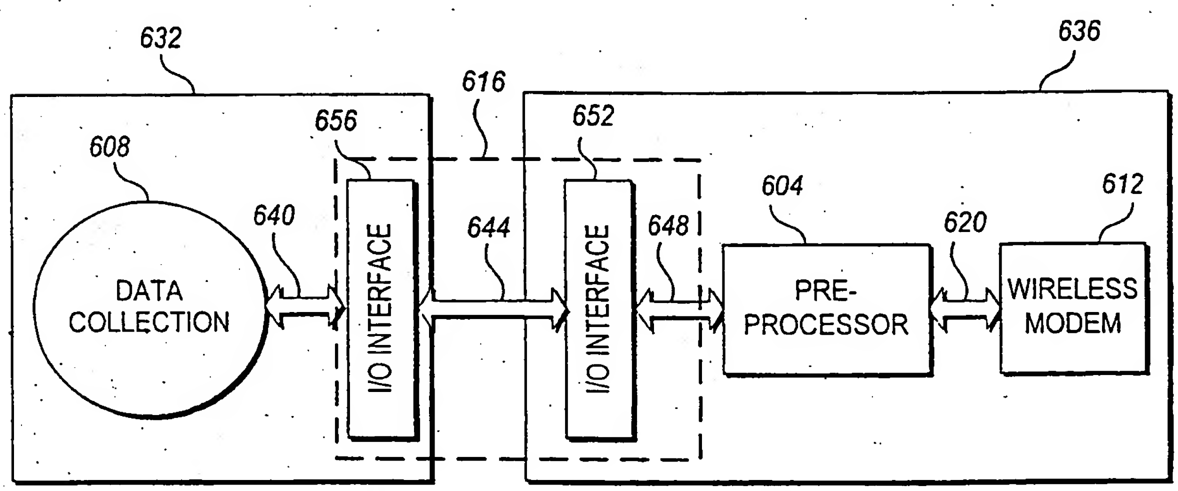


FIG. 6C

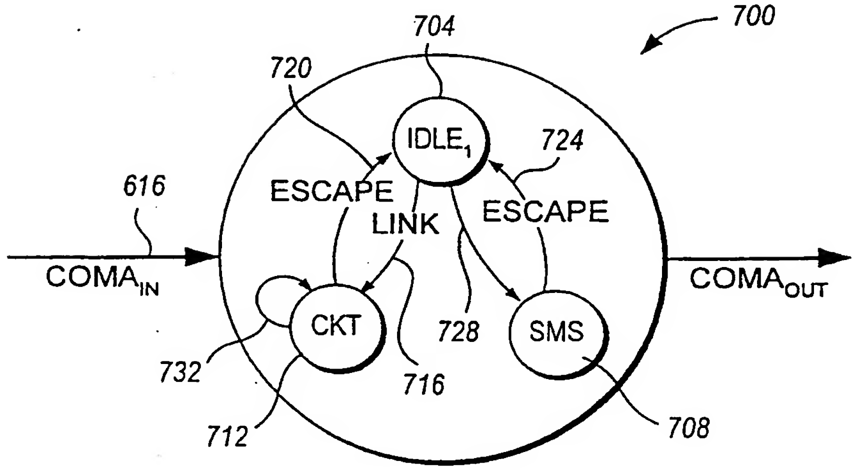

700
704
720
IDLE$_1$
724
616
ESCAPE
LINK
ESCAPE
COMA$_{IN}$
COMA$_{OUT}$
CKT
728
SMS
732
716
712
708

FIG. 7A

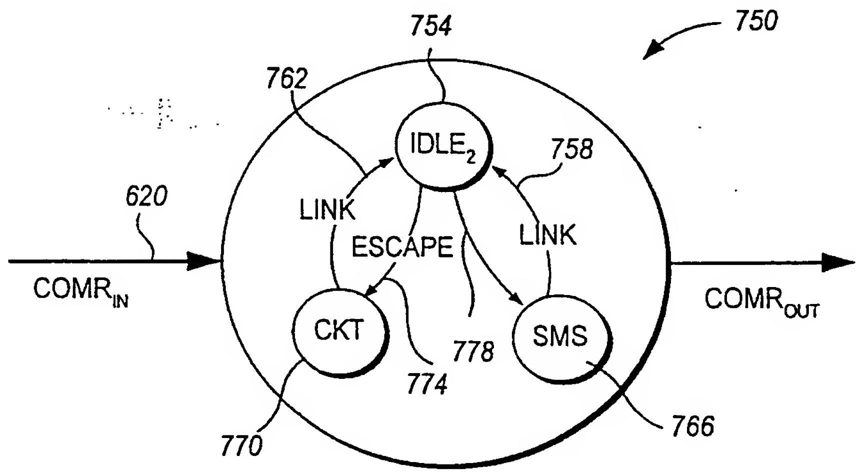

750
754
762
IDLE$_2$
758
620
LINK
ESCAPE
LINK
COMR$_{IN}$
COMR$_{OUT}$
CKT
778
SMS
774
770
766

FIG. 7B

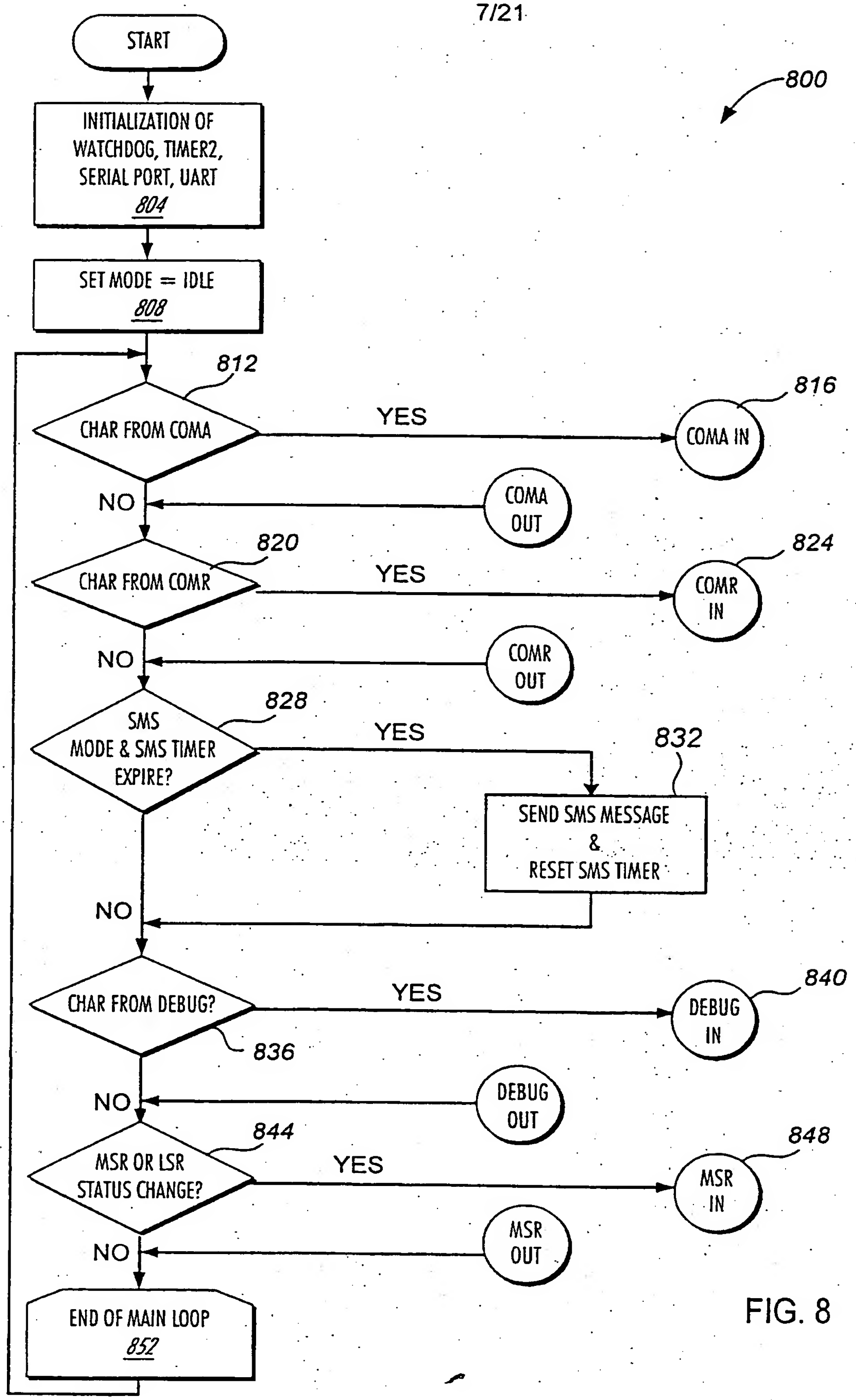

800
START
INITIALIZATION OF WATCHDOG, TIMER2, SERIAL PORT, UART
804
SET MODE = IDLE
808
812
CHAR FROM COMA
YES
COMA IN
816
NO
COMA OUT
820
CHAR FROM COMR
YES
COMR IN
824
NO
COMR OUT
828
SMS MODE & SMS TIMER EXPIRE?
YES
832
SEND SMS MESSAGE & RESET SMS TIMER
NO
CHAR FROM DEBUG?
836
YES
DEBUG IN
840
NO
DEBUG OUT
844
MSR OR LSR STATUS CHANGE?
YES
MSR IN
848
NO
MSR OUT
END OF MAIN LOOP
852

FIG. 8

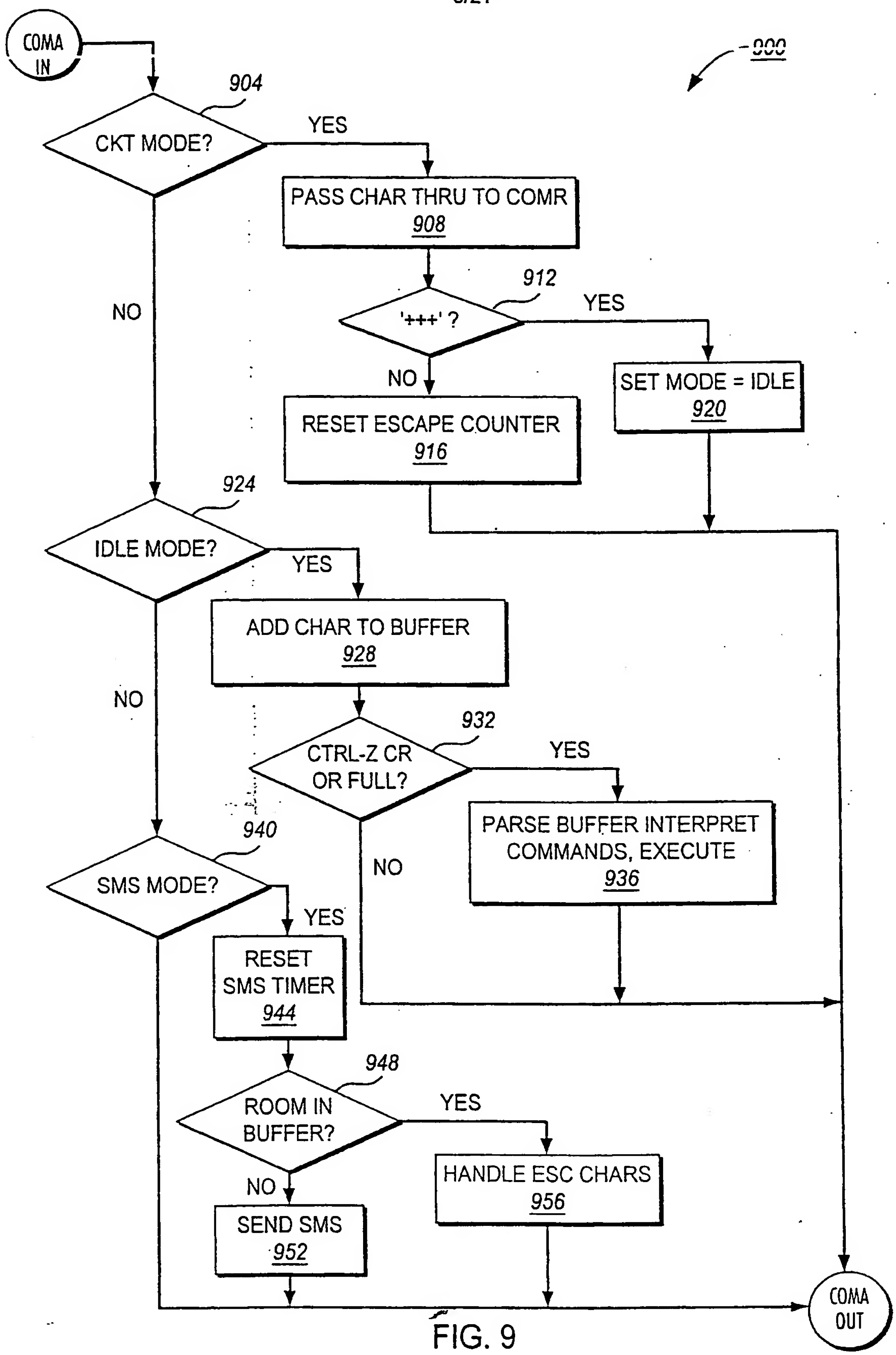

900
COMA IN
904
CKT MODE?
YES
NO
PASS CHAR THRU TO COMR
908
912
'+++' ?
YES
NO
SET MODE = IDLE
920
RESET ESCAPE COUNTER
916
924
IDLE MODE?
YES
NO
ADD CHAR TO BUFFER
928
932
CTRL-Z CR OR FULL?
YES
NO
PARSE BUFFER INTERPRET COMMANDS, EXECUTE
936
940
SMS MODE?
YES
RESET SMS TIMER
944
948
ROOM IN BUFFER?
YES
NO
HANDLE ESC CHARS
956
SEND SMS
952
COMA OUT

FIG. 9

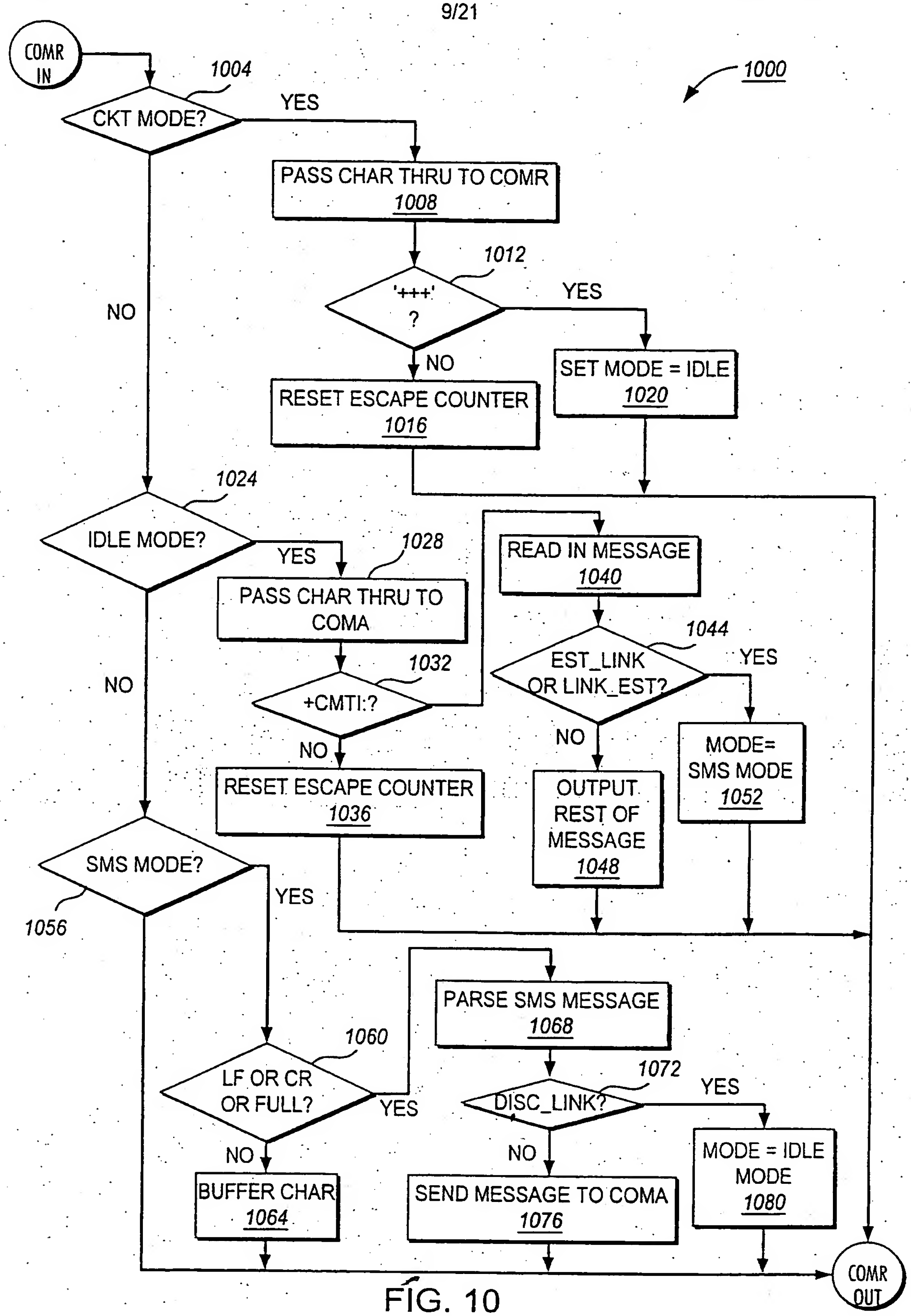

1000
COMR IN
1004
CKT MODE?
YES
NO
PASS CHAR THRU TO COMR
1008
1012
'+++' ?
YES
NO
SET MODE = IDLE
1020
RESET ESCAPE COUNTER
1016
1024
IDLE MODE?
YES
NO
1028
PASS CHAR THRU TO COMA
1032
+CMTI:?
NO
RESET ESCAPE COUNTER
1036
READ IN MESSAGE
1040
1044
EST_LINK OR LINK_EST?
YES
NO
MODE= SMS MODE
1052
OUTPUT REST OF MESSAGE
1048
SMS MODE?
1056
YES
1060
LF OR CR OR FULL?
YES
NO
BUFFER CHAR
1064
PARSE SMS MESSAGE
1068
1072
DISC_LINK?
YES
NO
SEND MESSAGE TO COMA
1076
MODE = IDLE MODE
1080
COMR OUT

FIG. 10

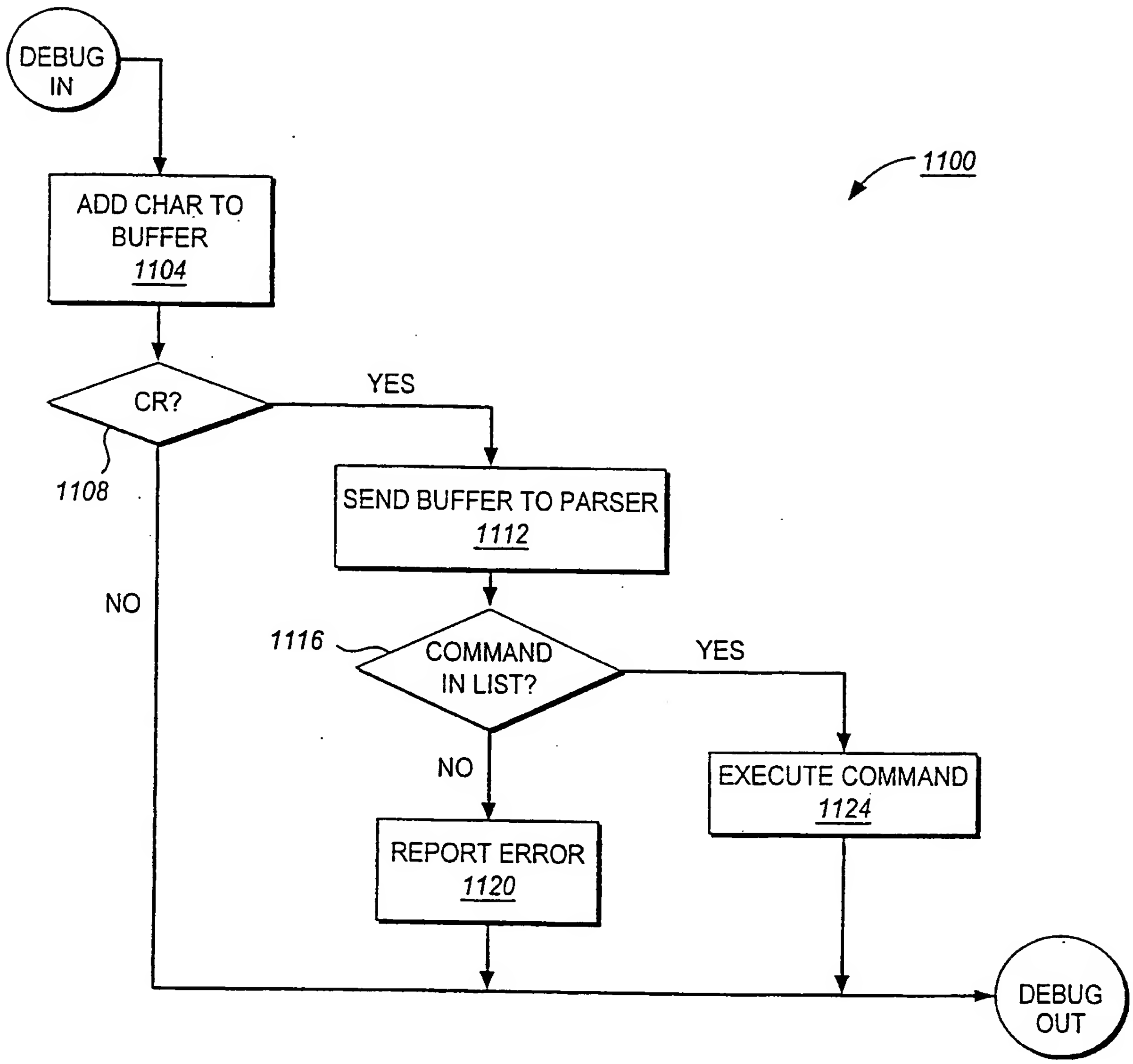

DEBUG IN
ADD CHAR TO BUFFER 1104
CR?
1108
YES
NO
SEND BUFFER TO PARSER 1112
1116
COMMAND IN LIST?
YES
NO
EXECUTE COMMAND 1124
REPORT ERROR 1120
DEBUG OUT
1100

FIG. 11

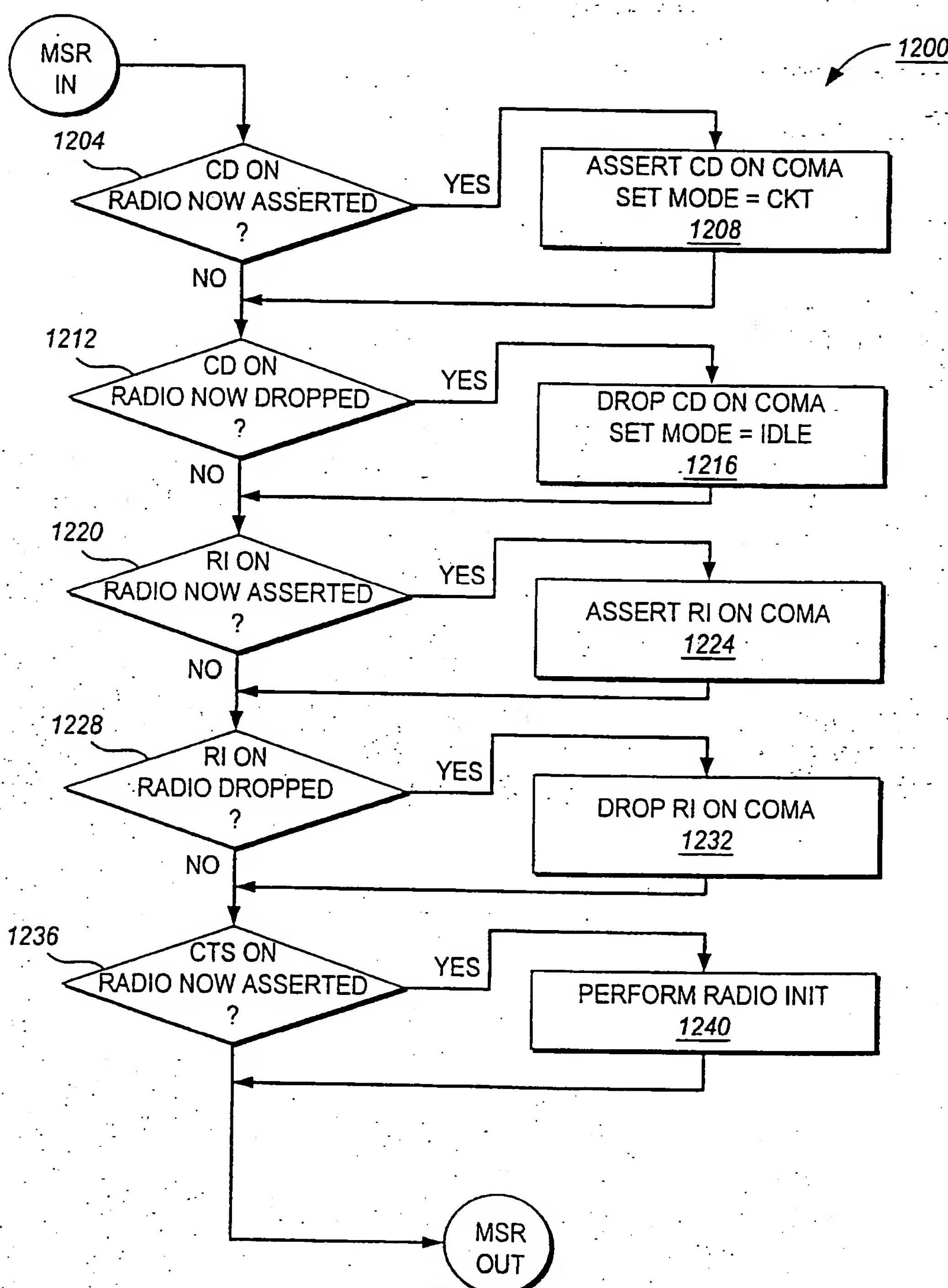

1200
MSR IN
1204
CD ON RADIO NOW ASSERTED ?
YES
ASSERT CD ON COMA SET MODE = CKT 1208
NO
1212
CD ON RADIO NOW DROPPED ?
YES
DROP CD ON COMA SET MODE = IDLE 1216
NO
1220
RI ON RADIO NOW ASSERTED ?
YES
ASSERT RI ON COMA 1224
NO
1228
RI ON RADIO DROPPED ?
YES
DROP RI ON COMA 1232
NO
1236
CTS ON RADIO NOW ASSERTED ?
YES
PERFORM RADIO INIT 1240
MSR OUT

FIG. 12

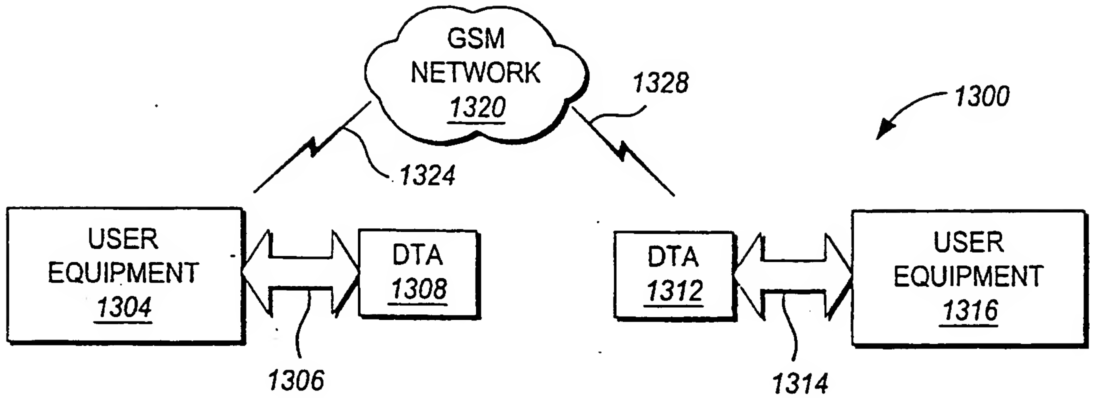


FIG. 13A

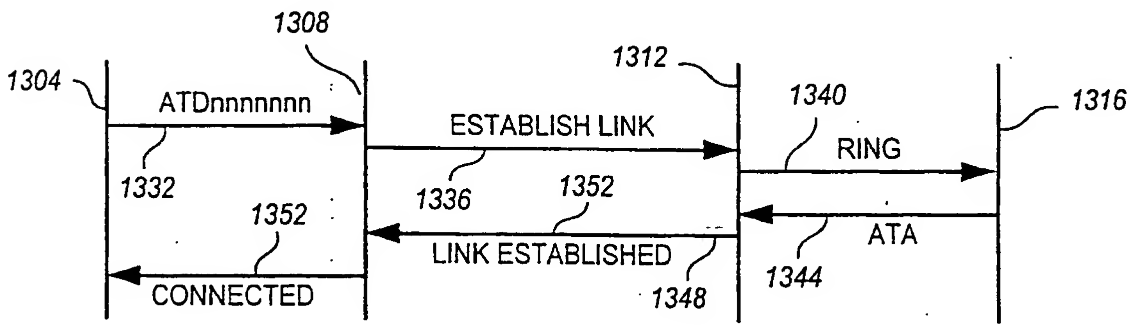


FIG. 13B

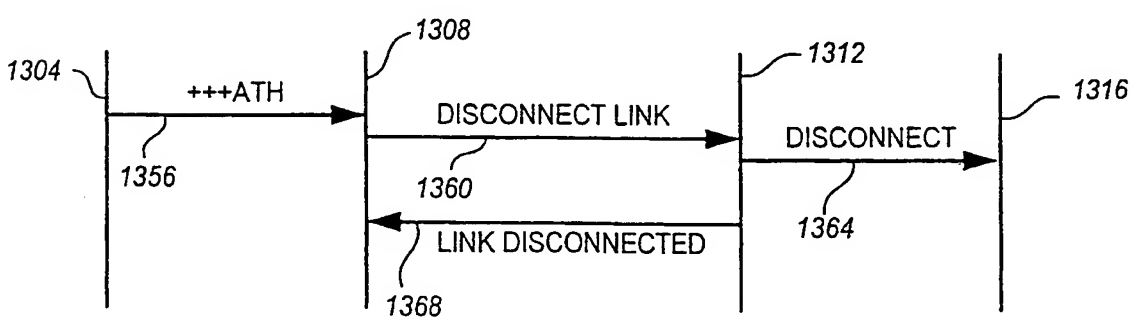


FIG. 13C

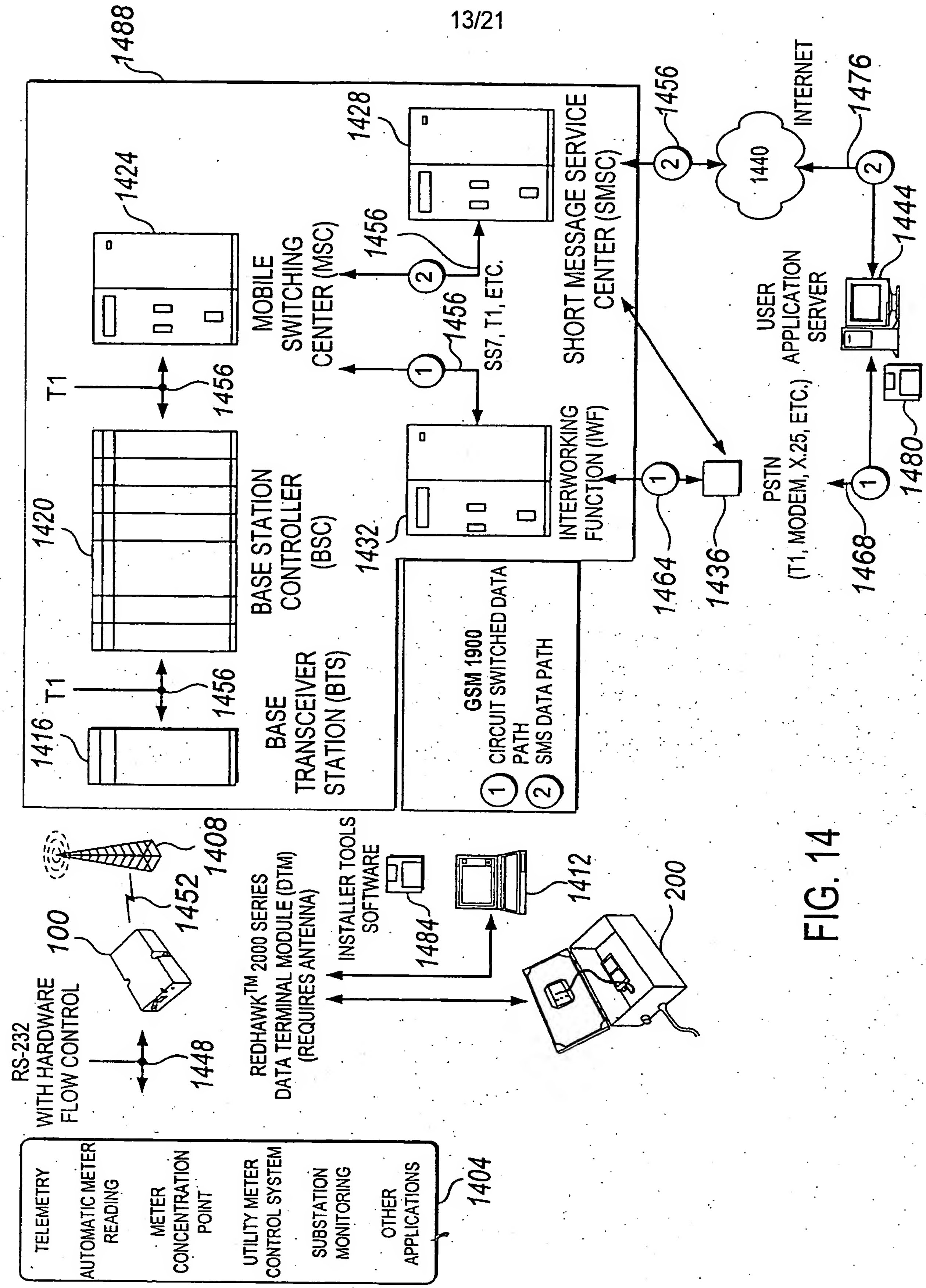

1488
1416
1420
1424
1428
1432
1456
T1
BASE TRANSCEIVER STATION (BTS)
BASE STATION CONTROLLER (BSC)
MOBILE SWITCHING CENTER (MSC)
SS7, T1, ETC.
INTERWORKING FUNCTION (IWF)
SHORT MESSAGE SERVICE CENTER (SMSC)
GSM 1900
1 CIRCUIT SWITCHED DATA PATH
2 SMS DATA PATH
1464
1436
1440
INTERNET
1476
1444
USER APPLICATION SERVER
1480
PSTN (T1, MODEM, X.25, ETC.)
1468
TELEMETRY
AUTOMATIC METER READING
METER CONCENTRATION POINT
UTILITY METER CONTROL SYSTEM
SUBSTATION MONITORING
OTHER APPLICATIONS
1404
RS-232 WITH HARDWARE FLOW CONTROL
1448
100
1452
1408
REDHAWK™ 2000 SERIES DATA TERMINAL MODULE (DTM) (REQUIRES ANTENNA)
INSTALLER TOOLS SOFTWARE
1484
1412
200

FIG. 14

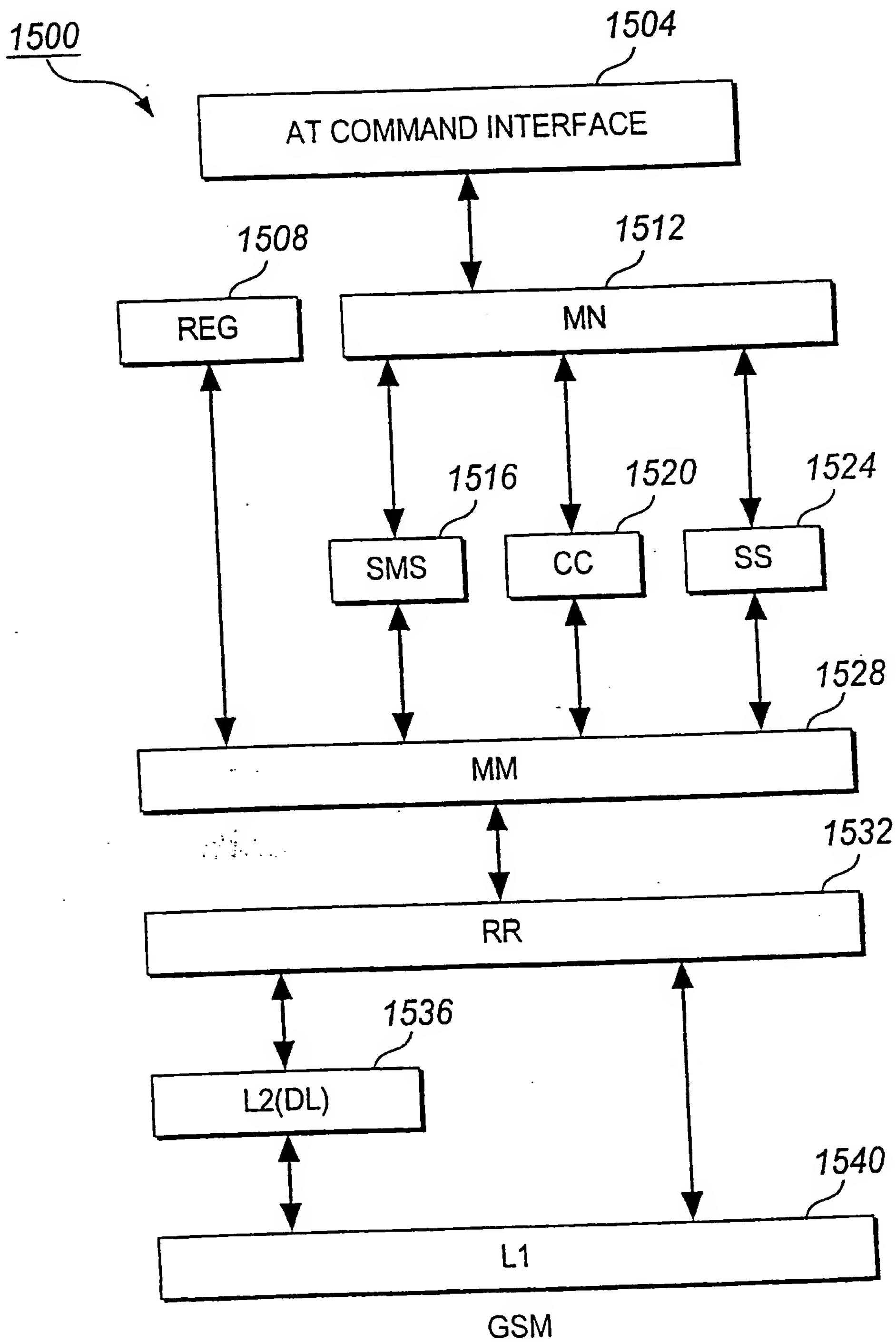
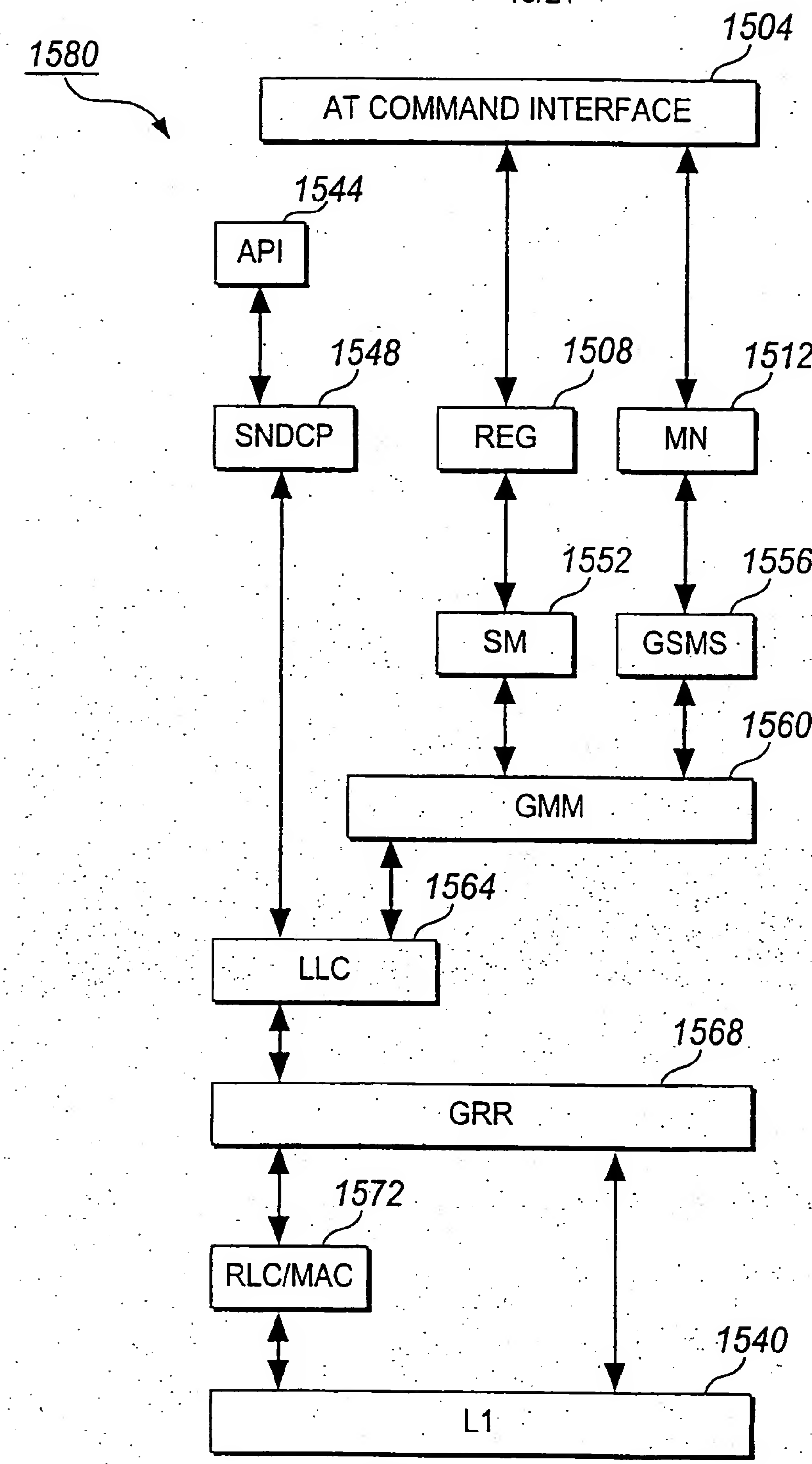
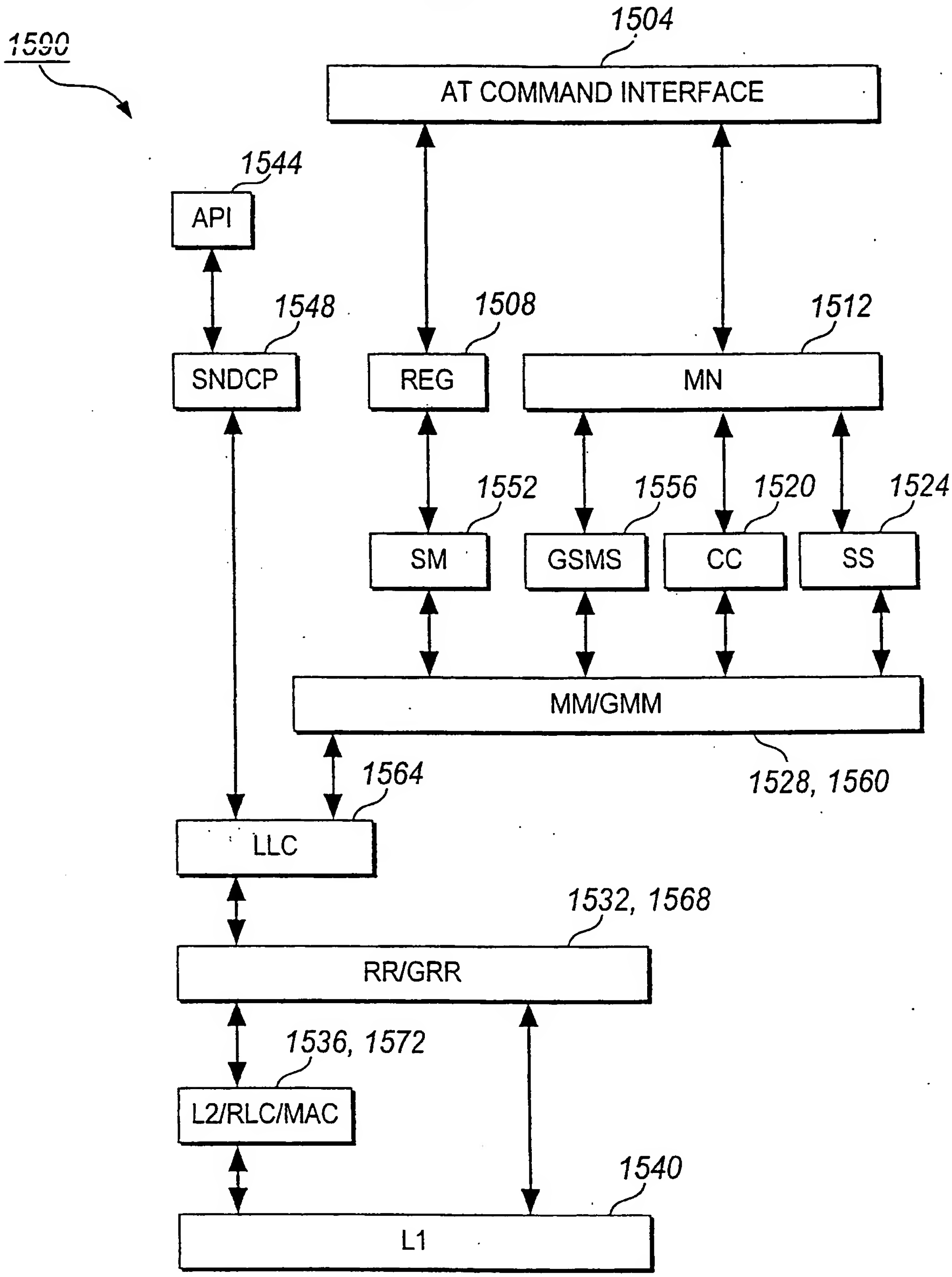

1500
1504
AT COMMAND INTERFACE
1508
REG
1512
MN
1516
SMS
1520
CC
1524
SS
1528
MM
1532
RR
1536
L2(DL)
1540
L1
GSM

FIG. 15A



GPRS CLASS C

FIG. 15B



GPRS CLASS A

FIG. 15C

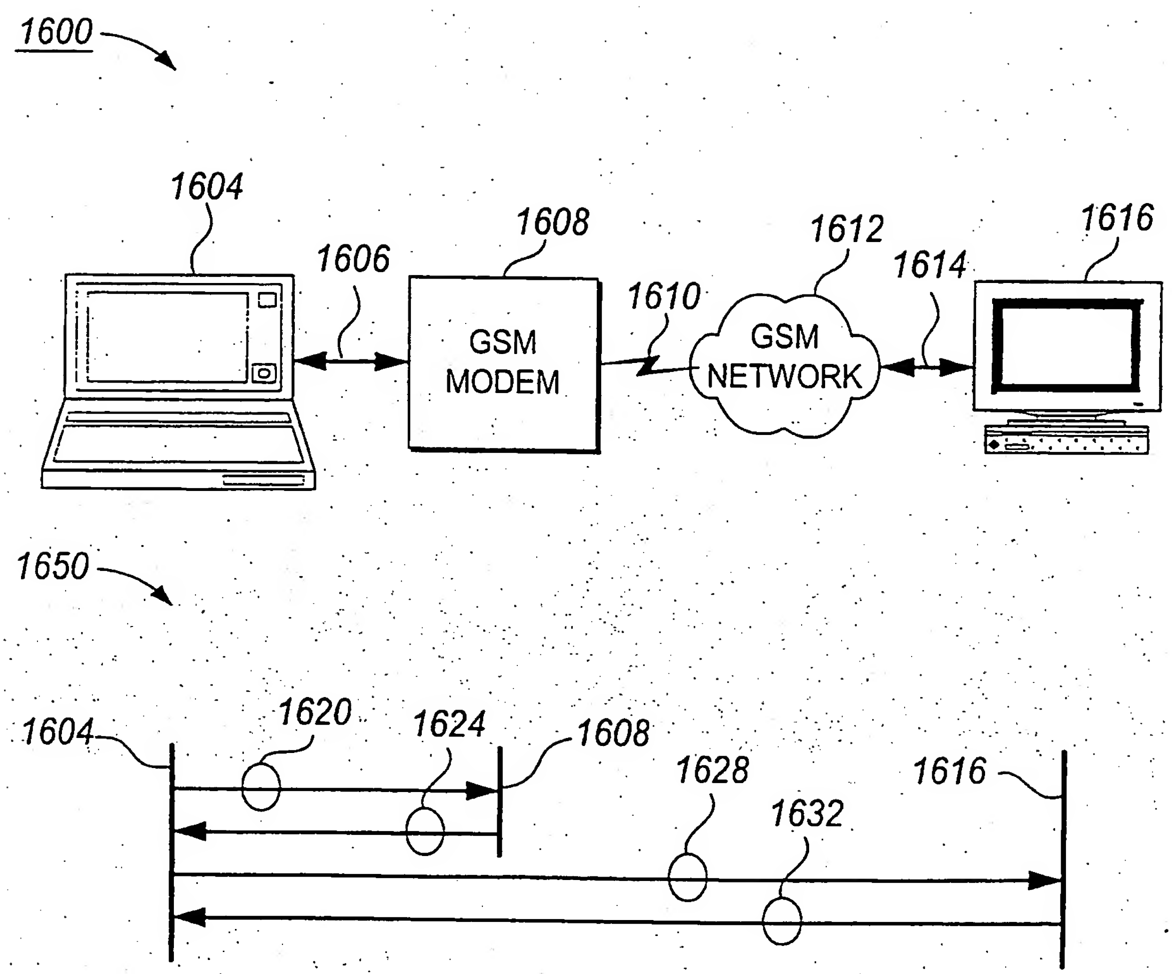

1600
1604
1606
1608
GSM MODEM
1610
1612
GSM NETWORK
1614
1616
1650
1604
1620
1624
1608
1628
1632
1616

FIG. 16

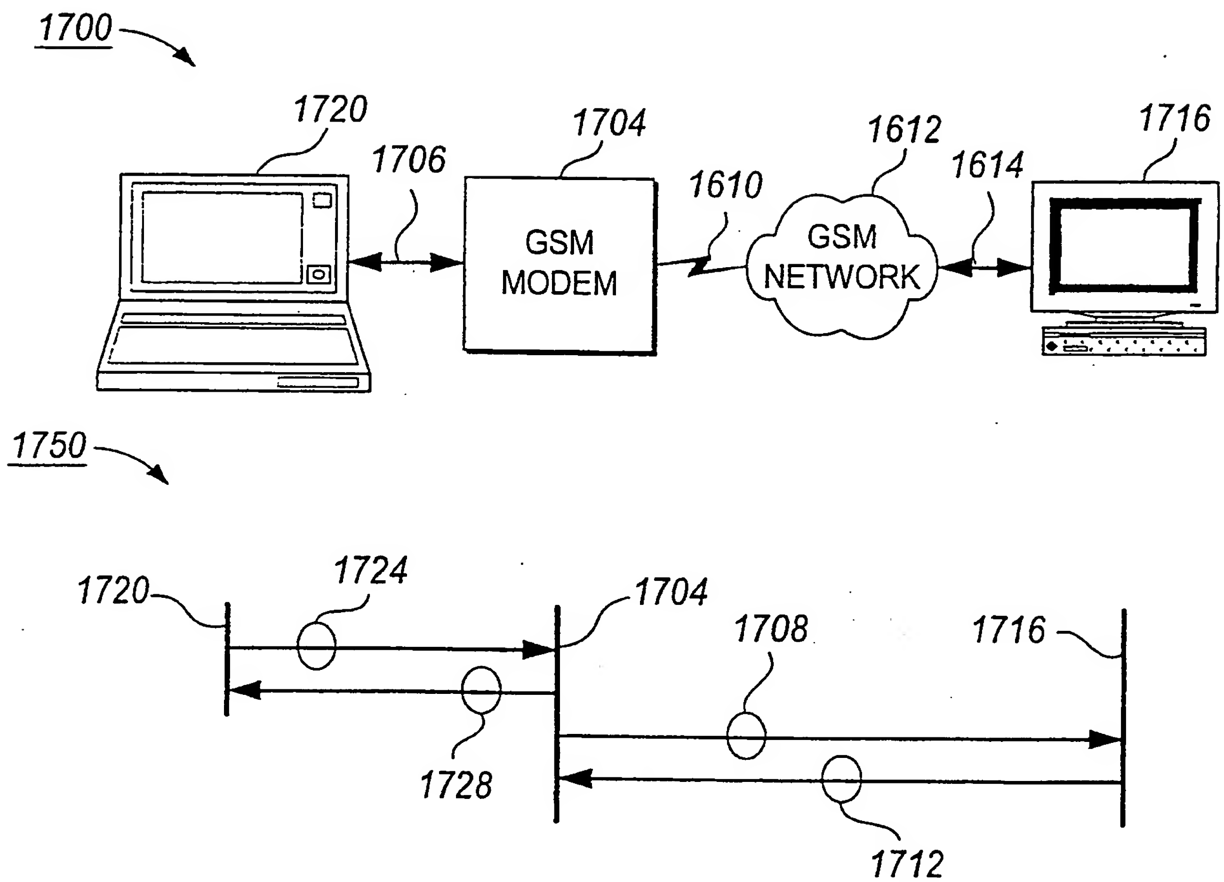

1700
1720
1706
1704
GSM
MODEM
1610
1612
GSM
NETWORK
1614
1716
1750
1720
1724
1704
1708
1716
1728
1712

FIG. 17

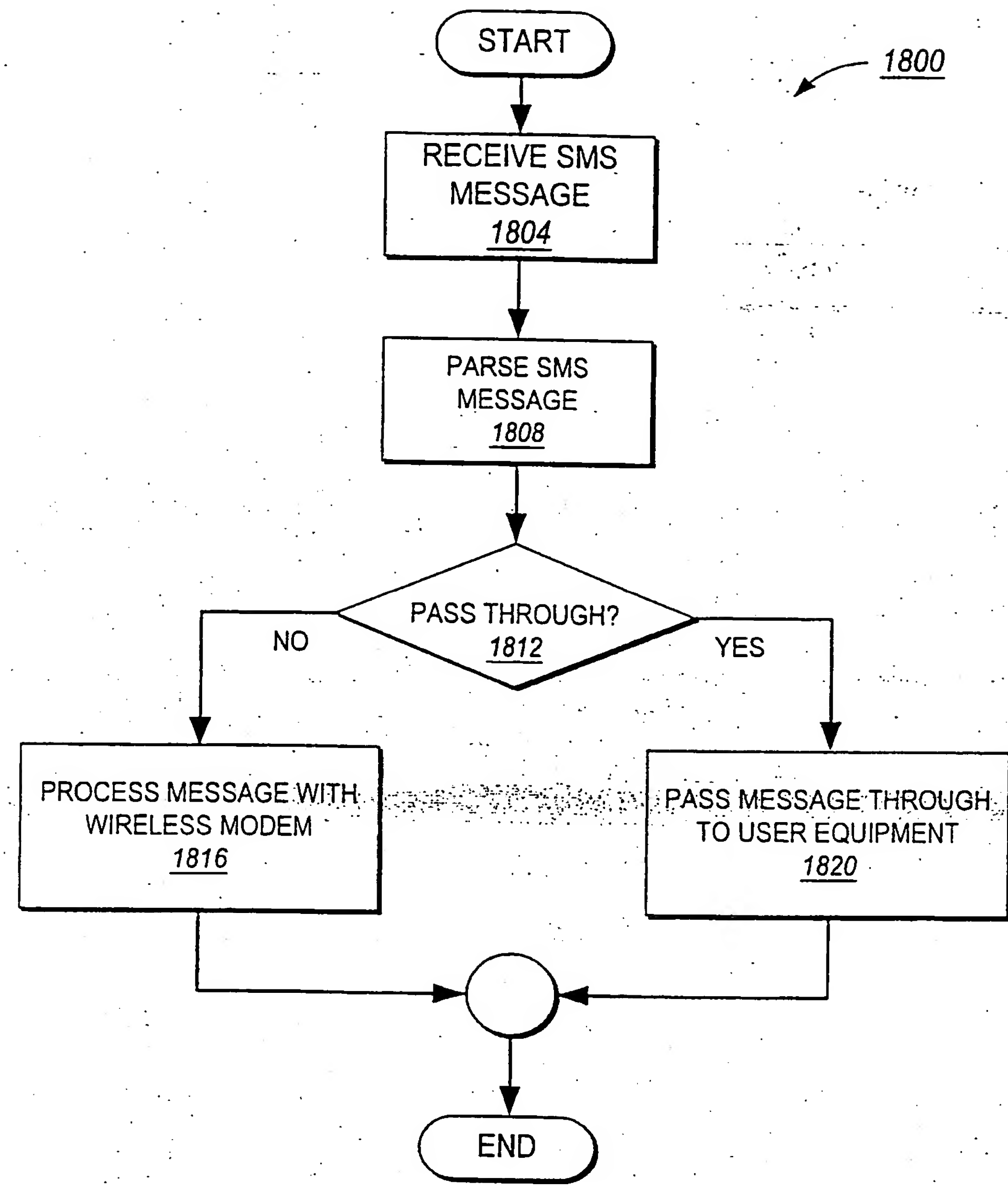

START
1800
RECEIVE SMS MESSAGE 1804
PARSE SMS MESSAGE 1808
PASS THROUGH? 1812
NO
YES
PROCESS MESSAGE WITH WIRELESS MODEM 1816
PASS MESSAGE THROUGH TO USER EQUIPMENT 1820
END

FIG. 18

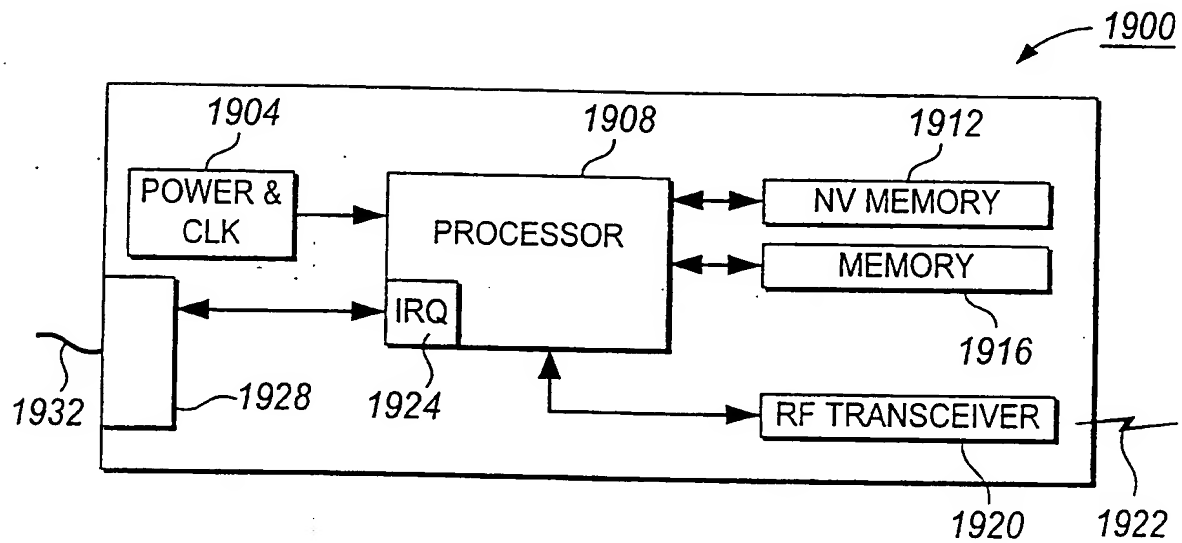


FIG. 19A

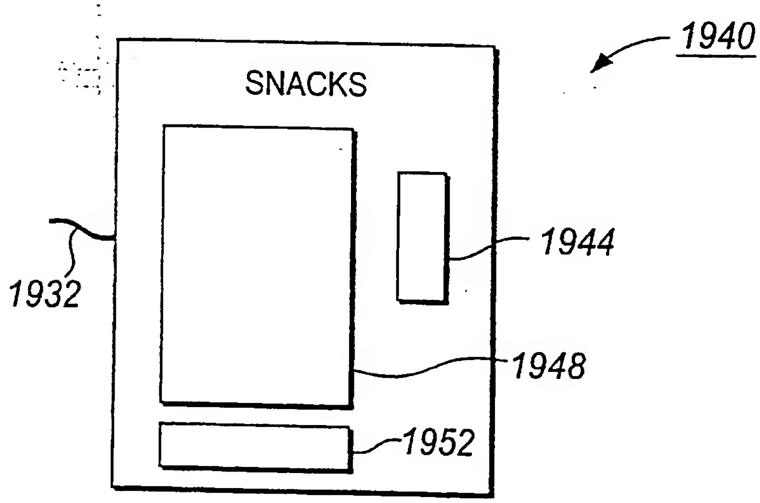


FIG. 19B

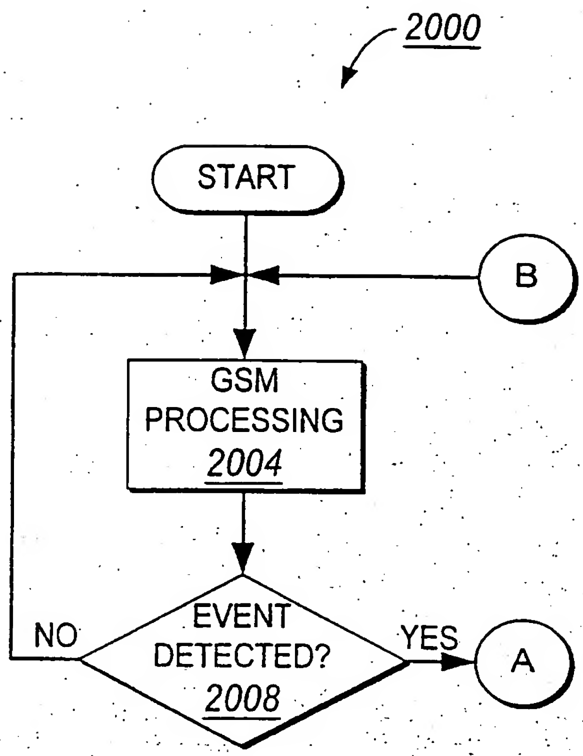

2000
START
B
GSM
PROCESSING
2004
EVENT
DETECTED?
2008
NO
YES
A

FIG. 20

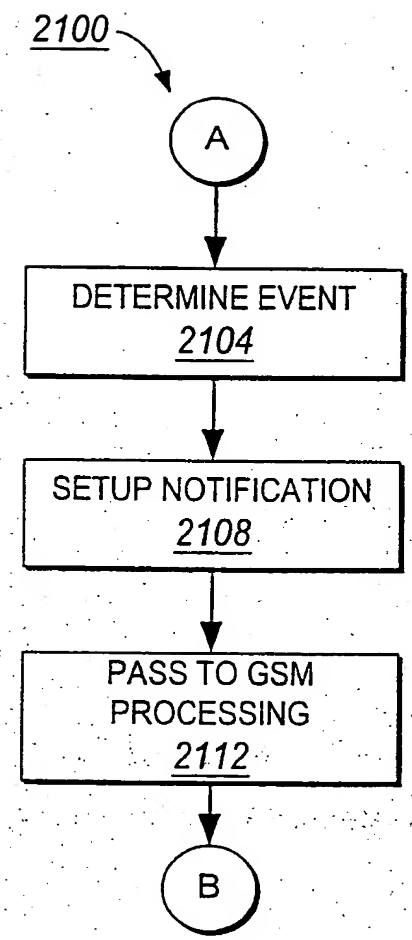

2100
A
DETERMINE EVENT
2104
SETUP NOTIFICATION
2108
PASS TO GSM
PROCESSING
2112
B

FIG. 21

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US00/25808

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) :G06F 17/50, 9/455; G06G 7/62
US CL :703/13, 23
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 703/13, 23

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EAST, IEEE Xplore, Proquest
search terms: modem, GSM, simulat* or model* or emulat*

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	MORASSI et al. Real Time Simulation of Fax Transmission on the Transparent GSM Data Service. 5th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications. Pages 944-947, especially Fig. 1.	1-3, 5-7, 9-11
X	EP 0849682 A1 (JACOBER et al) 24 June 1998, col. 4, lines 3-14, and Fig. 1.	1-3, 5-7, 9-11
Y — A	US 5,559,870 A (PATTON et al) 24 September 1996, col. 2 line 65 through col. 3 line 67, and Fig. 1.	8 — 1-7, 9-11
Y — A	BOLTON. C.R. The Cellular Way to Retrieve Data. Transmission & Distribution. Octobeı 1994. Pages 54, 57.	8 — 1-7, 9-11

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search 07 JANUARY 2001	Date of mailing of the international search report 31 JAN 2001
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer KEVIN TESKA James R. Matthews Telephone No. (703) 305-9704

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y — A	MURCH. A. et al. Cellular Data Services Over GSM. Telecommunications. March 1995. Pages 63-64.	8 — 1-7, 9-11
A	ANONYMOUS. GSM Modules for Remote Applications. Electronic Times. 9 November 1998.	1-11
A	ORR. S. Cell Phones Act as Modems. Computer Dealer News. 14 May 1999.	1-11
A	US 5,537,458 A (SUOMI et al) 16 July 1996.	1-11

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US00/25808

Box I Observations where certain claims were found unsearchable (Continuation of item 1 of first sheet)

This international report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box II Observations where unity of invention is lacking (Continuation of item 2 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

Please See Extra Sheet.

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.

2. ☐ As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.

3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☒ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:
1-11

Remark on Protest ☐ The additional search fees were accompanied by the applicant's protest.
☐ No protest accompanied the payment of additional search fees.

BOX II. OBSERVATIONS WHERE UNITY OF INVENTION WAS LACKING
This ISA found multiple inventions as follows:

This application contains the following inventions or groups of inventions which are not so linked as to form a single inventive concept under PCT Rule 13.1. In order for all inventions to be searched, the appropriate additional search fees must be paid.

Group I, claim(s) 1-11, drawn to a communications bridge simulator and a method for simulating a circuit switched call link.
Group II, claim(s) 12-26, drawn to a modem with modem mangement information messages and a method of modem mangement.
Group III, claim(s) 27-44, drawn to a modem with event detection of alarm signals and a method of event detection.

The inventions listed as Groups I-III do not relate to a single inventive concept under PCT Rule 13.1 because, under PCT Rule 13.2, they lack the same or corresponding special technical features for the following reasons:

Groups II and III lack the special techical feature of Group I comprising the simulation of a circuit switched call link.

Groups I and III lack the special technical feature of Group II comprising the handling of short message service messages comprising modem management information.

Groups I and II lack the special technical feature of Group III comprising the handling of an alarm signal using an event dectection interface.